

Verfahren zur Simulation und Analyse der Auswirkungen toleranzbedingter Bauteilabweichungen

Der Technischen Fakultät
der Universität Erlangen-Nürnberg
zur Erlangung des Grades

Doktor-Ingenieur

vorgelegt von
Stefan Wittmann

Erlangen 2011

Als Dissertation genehmigt von
der Technischen Fakultät der
Universität Erlangen-Nürnberg

Tag der Einreichung:	27. September 2010
Tag der Promotion:	01. Februar 2011
Dekan:	Prof. Dr.-Ing. habil R. German
Berichterstatter:	Prof. Dr.-Ing. H. Meerkamm Prof. Dr.-Ing. M. Stamminger

Danksagung

Diese Arbeit entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter am Lehrstuhl für Konstruktionstechnik im Zeitraum 2005 bis 2010 an der Friedrich-Alexander-Universität Erlangen-Nürnberg. Beeinflusst und unterstützt wurde ich in dieser Zeit von zahlreichen Personen, bei denen ich mich an dieser Stelle bedanken möchte.

Prof. Dr.-Ing. H. Meerkamm, dem ehemaligen Inhaber des Lehrstuhls für Konstruktionstechnik und Hauptgutachter der Dissertation danke ich für die Betreuung, das entgegengebrachte Vertrauen und den Freiraum, den er mir bei der Forschung ermöglichte. Prof. Dr.-Ing. Marc Stamminger danke ich für die Übernahme des Koreferats und für das Interesse an den bearbeiteten Themenstellungen.

Weiterhin danke ich den Mitarbeitern der Toleranzgruppe am Lehrstuhl für Konstruktionstechnik für die Diskussion der vorgestellten Ideen aus unterschiedlichsten Blickwinkeln und die zahlreichen Anregungen. Insbesondere danke ich hierbei Andreas Stockinger und Tobias Stoll für die erfolgreiche enge Zusammenarbeit.

Einen großen Beitrag zum Gelingen der Dissertation leisteten Studien- und Diplomarbeiter sowie wissenschaftliche Hilfskräfte, insbesondere Florian Penzkofer und Alexandr Gnezdilov.

Weiterhin danke ich meinen Kooperationspartnern an Lehrstühlen der technischen Fakultät Sabine Helwig (ehem. Lehrstuhl Hardware-Software-Co-Design), Marco Winter und Gerd Sussner (beide ehem. Lehrstuhl für graphische Datenverarbeitung), Klaus Iglberger (Lehrstuhl für Systemsimulation) und Gabriella Kókai (ehem. Lehrstuhl für Programmiersysteme). Seitens der Industriepartner gilt mein besonderer Dank Stefan Gliniorz (Bosch GmbH) für die motivierende Zusammenarbeit und das wertvolle Feedback aus der Praxis.

Last but not least danke ich meiner Familie, meinem Großvater Johann Eber und meiner Verlobten Sabine Helwig, die durch ihren Rückhalt und ihre Unterstützung den Abschluss dieser Arbeit erst ermöglicht haben.

Inhaltsverzeichnis

1	Einleitung.....	1
1.1	Problemstellung	1
1.2	Aufgabenstellung	2
1.3	Tolerierung von Bauteilen	2
1.3.1	Bauteilabweichungen.....	3
1.3.2	Toleranzvergabe in technischen Zeichnungen.....	3
1.3.3	Tolerierungsgrundsätze.....	7
1.3.4	Auswirkungen der Toleranzvergabe	8
1.4	Grundlagen der verwendeten Computergrafik-Techniken	10
1.4.1	Geometrische Modellierung in CAD-Systemen	10
1.4.2	Dreiecksnetze.....	13
1.4.3	Kollisionsberechnung	15
1.4.4	Transformationen.....	19
2	Stand der Forschung und Technik (Toleranzanalyse)	22
2.1	Arithmetische Toleranzanalyse.....	22
2.2	Statistische Toleranzanalyse	23
2.2.1	Root Sum Squares Methode	24
2.2.2	Monte-Carlo-Methode	26
2.2.3	Taguchi-Methode und Erweiterungen	26
2.2.4	Beitragsleisteranalyse	26
2.3	Modelle zur Simulation der Auswirkungen von Abweichungen.....	27
2.3.1	Vereinfachende Modelle zur dreidimensionalen Abweichungsbeschreibung	27
2.3.2	Bestehende geometrieorientierte Analyseverfahren	30
2.4	Umsetzung in kommerziellen Systemen.....	40
3	Geometrieorientierte Toleranzanalyse: Motivation und Überblick	41
3.1	Ableitung des Handlungsbedarfs	41
3.2	Erstellung nichtidealer Bauteile.....	42
3.2.1	Einzelteilerzeugung	42
3.2.2	Wahl des Datenformats für die Geometrirepräsentation.....	43
3.3	Baugruppenerzeugung	45
3.3.1	Relative Positionierung.....	45

3.3.2	Pfadplanung	45
3.3.3	Monte-Carlo-Ansatz	46
3.4	Auswertungsverfahren zur Analyse abweichungsbehafteter Baugruppen	46
3.4.1	Analyse einzelner Baugruppen	46
3.4.2	Statistische Analyse durch Volumenvisualisierung.....	47
3.5	Zusammenfassung der Geometrieorientierten Toleranzanalyse.....	47
4	Relative Positionierung.....	50
4.1	Heuristische Optimierung	51
4.2	Positionierungsframework	53
4.3	Bewertungsfunktionen	54
4.3.1	Kollisionsvermeidung.....	54
4.3.2	Aufsummierter quadratischer Abstand (SSD)	55
4.3.3	Hausdorff-Abstand.....	56
4.3.4	Richtungsbewertung	56
4.4	Verfahren zur Beschleunigung der Positionierung.....	58
4.4.1	Vorzeitiger Berechnungsabbruch beim Aufsummieren	58
4.4.2	Abschätzen der Bewertung durch Auswahl weniger Vertices.....	58
4.4.3	Parallelisierung	59
4.5	Experimentelle Ergebnisse	60
4.5.1	Zero-Test Hausdorff-Abstand.....	60
4.5.2	Korrelation zwischen vollständigem und abgeschätztem Hausdorffabstand	61
4.5.3	Positionierung eines Kreuzziehteils auf einer ebenen Fläche.....	63
4.5.4	Zusammenbau zweier nominaler Kreuzziehteile.....	65
4.5.5	Relative Positionierung simulierter Kreuzzieh-Bauteile	66
4.6	Fazit und Ausblick Relative Positionierung	70
5	Pfadplanung	72
5.1	Betrachtung bestehender Verfahren.....	73
5.1.1	Grundbegriffe der Pfadplanung	73
5.1.2	Bestehende Ansätze	75
5.2	Besondere Herausforderungen bei der Analyse toleranzbehafteter Bauteile ..	80
5.3	Entwickelter Pfadplanungsansatz zur statistischen Toleranzanalyse	80
5.3.1	Verwenden eines Hilfspfades	81
5.3.2	Einschränkung des Suchraums	82

5.3.3	Angepasstes Sampling des Suchraumes	83
5.4	Experimentelle Ergebnisse	84
5.4.1	Vergleich bestehender Verfahren	84
5.4.2	Einbinden eines Hilfspfades	91
5.4.3	Hilfspfad mit Suchraumanpassung	92
5.4.4	Verwendung angepasster Samplingverfahren	92
5.4.5	Kombination aller entwickelten Verfahren.....	93
5.5	Pfadplanung: Zusammenfassung und Ausblick.....	95
6	Analyseverfahren zur Auswertung von Abweichungen.....	97
6.1	Graphische Analyse einzelner Baugruppen durch Abstandsmessungen	97
6.1.1	Definition von Abstandsmessungen	97
6.1.2	Auswertung am Beispiel der Napf-Baugruppe.....	99
6.2	Haptische Analyse von nichtidealen Baugruppen	102
6.2.1	Stand der Technik	102
6.2.2	Zielsetzungen	103
6.2.3	Aufbau des entwickelten haptischen Analysesystems.....	105
6.2.4	Untersuchung der Genauigkeit des entwickelten Systems	107
6.2.5	Kombinierte Spalt- und Versatzanalyse	111
6.2.6	Montagesimulation nichtidealer Bauteile	113
6.2.7	Haptische Analyse: Zusammenfassung und Ausblick.....	114
6.3	Statistische Analyse durch Volumenvisualisierung.....	115
6.3.1	Idee.....	115
6.3.2	Erzeugen eines statistischen Datensatzes: Summierte Voxelisierung ..	116
6.3.3	Ergebnisvisualisierung.....	119
6.3.4	Volumenbasierte statistische Kollisionserkennung	120
6.3.5	Experimentelle Ergebnisse	123
6.3.6	Volumenvisualisierung: Zusammenfassung und Ausblick.....	129
7	Zusammenfassung und Ausblick.....	131
8	Literaturverzeichnis	135
9	Anhang.....	146
9.1	Prozessfähigkeitsindex C_p	146
9.2	Pfadvereinfachung	146
9.3	MPNN, Abstandsmetriken.....	147

Abkürzungsverzeichnis

AABB:	Axis Aligned Bounding Box
B-Rep:	Boundary Representation (Geometriemodell)
CAD:	Computer Aided Design
CAT:	Computer Aided Tolerancing
CDT:	Collision Detection Calls
CPU:	Central Processing Unit
CSG:	Constructive Solid Geometry (Geometriemodell)
CT:	Computertomographie
DOF:	Degrees of Freedom
EA:	Evolutionärer Algorithmus
FE:	Finite Elemente
FFD:	Free-Form Deformation
HSV:	Hue Saturation Value (Farbraum)
ICP:	Iterative Closest Point Algorithmus
NURBS:	Non-Uniform Rational B-Spline
OBB:	Oriented Bounding Box
ODE:	Open Dynamics Engine
PE:	Physics Engine
PQP:	Proximity Query Package
PRM:	Probabilistic Roadmap Algorithmus
PRM-Vis:	Visibility Based PRM Algorithmus
PSO:	Particle Swarm Optimisation
RGB:	Red Green Blue (Farbraum)
RRT:	Rapidly Exploring Random Tree Algorithmus
RRT-Con:	RRT-Connect Algorithmus
RRT-ExtCon:	RRT-Extend-Connect Algorithmus
SA:	Simulated Annealing
SDT:	Small Displacement Torsor
SSD:	Summed Squared Distance
STL:	Standard Tessellation Language
TTRS:	Topologically and Technologically Related Surface
VRML:	Virtual Reality Modeling Language

M:	Matrix (Großbuchstabe, fett)
<i>s:</i>	Skalar aus R (Kleinbuchstabe, <i>kursiv</i>)
v:	Vektor aus R hoch n (Kleinbuchstabe, fett formatiert)
C_{free} :	Collision-free Space
C_{obs} :	Obstacle Space
C_p :	Prozessfähigkeitsindex, siehe Anhang
T_a :	Arithmetische Schließtoleranz
T_i :	Einzeltoleranz
T_q :	Quadratische Schließtoleranz
T_w :	Wahrscheinliche Schließtoleranz
δ :	Schrittweite des lokalen Planers
μ :	Mittelwert
σ :	Standardabweichung

Abstract

Manufactured parts always have geometric deviations compared to the designed ideal shape. Tolerances define, in which range these errors are acceptable. Because the tolerance definition causes complex spatial effects, simulation programs are used to support the analysis. The part geometry and the assembly process are strongly abstracted in common commercial systems. Therefore, many real problems can not be modelled or detected.

This thesis shows how the effects of tolerance-related deviations can be simulated and analysed based on detailed geometric modelling. Non-ideal part variants are generated as triangle meshes (i.e. through FE-simulation). Because of the geometric deviations of individual parts, a repositioning is necessary to build up a non-ideal assembly. For this positioning (also called relative positioning) a framework has been developed, which is based on heuristic optimization. The approach allows for the compliance to plausibility criteria (i.e. collision avoidance between parts) and the consideration of the positioning scheme.

Furthermore, new automatic path planning algorithms have been developed to check the assembly process of non-ideal part variants in a short computation time. For the analysis of resulting non-ideal assemblies, haptic methods and automatic distance measurements have been developed. These methods can be used to examine single non-ideal assemblies. To visualize the condensed result of statistical tolerance simulations, a method of summed voxelization has been developed. Based on a volume dataset, the geometry of all generated non-ideal assemblies is combined in a single scene.

The methods presented in this thesis can be used for the early prediction of the geometric effects of part defects. With a comparably low modelling effort, potential problems can be predicted before a physical prototype is manufactured. This way, non-ideal virtual prototypes potentially help to bring forward tolerance changes and design modifications to save overall costs and improve the quality of the sold product.

1 Einleitung

Durch den Einsatz von Computern hat sich in den letzten 30 Jahren die Arbeitsweise in der Produktentwicklung grundlegend verändert. Computer Aided Design-Systeme erlauben die schnelle Modellierung dreidimensionaler Bauteile und die Zusammensetzung zu komplexen Baugruppen. Neben der besseren Anschaulichkeit der Modellierung und der schnelleren Erstellung technischer Zeichnungen hat die Abbildung der Bauteile im Rechner einen weiteren entscheidenden Vorteil: schon vor dem ersten gefertigten Prototyp können umfangreiche Untersuchungen der Bauteile durchgeführt werden. Ziel der Anwendung von Simulationsverfahren ist es, frühzeitig viele Aspekte des Produkts (z. B. Fertigbarkeit, Montierbarkeit und Funktionserfüllung) zu überprüfen, um so bereits im Vorfeld Probleme zu erkennen und Fehler zu vermeiden.

1.1 Problemstellung

Ein wichtiger Simulationsbereich ist die computerunterstützte Toleranzsimulation (Computer-Aided Tolerancing, CAT). Durch Toleranzsimulationen wird überprüft, ob Fertigungsabweichungen der einzelnen Bauteile in Summe zu Problemen führen können. Bei der Analyse bestehender Toleranzanalysesysteme hat sich gezeigt, dass der vom Benutzer zu leistende Aufwand zur Modellierung und Überprüfung von Baugruppen momentan enorm hoch ist. Da die Einarbeitung in den Umgang mit diesen Werkzeugen sehr aufwändig ist, wird die Toleranzanalyse in Firmen meist von Experten oder eigenen Simulationsabteilungen durchgeführt. Eine direkte Analyse durch die jeweiligen Produktentwickler ist nicht durchführbar.

Die Bauteilgeometrie und die Montagevorgänge werden in den am Markt verbreiteten Systemen stark vereinfacht modelliert, wodurch sich viele in der Realität auftretende Probleme nicht abbilden und aufdecken lassen.

Weiterhin liefern Toleranzanalysesysteme nach dem Stand der Technik v.a. statistische Aussagen zu Messwertverteilungen. Die Ergebnisse der Simulation bedürfen anschließend einer Interpretation durch den Anwender, die von ihm ein hohes Maß an Erfahrung und räumlichem Vorstellungsvermögen erfordern. Eine Darstellung der tatsächlichen geometrischen Zusammenhänge und entstehenden Probleme war bisher aufgrund der Abstraktion nicht möglich.

1.2 Aufgabenstellung

Ziel dieser Arbeit ist daher die Entwicklung neuer Methoden zur Analyse von abweichungsbehafteten Bauteilen, die schnell zu aussagekräftigen und verlässlichen Ergebnissen führen. Die Arbeit verfolgt den Ansatz, die abweichungsbehafteten Bauteile detailliert abzubilden und auf Vereinfachungen von Geometrie und Montageprozessen so weit wie möglich zu verzichten. Voraussetzung für die Umsetzung dieses Ansatzes ist zum einen die stark gestiegene verfügbare Rechenleistung, die eine Weiterverarbeitung und Analyse von derart aufwändigen Modellen erst ermöglicht, zum anderen wurden einige in anderer Form oder mit anderer Zielsetzung bereits bestehende Verfahren der Informatik auf noch nicht betrachtete Problemstellungen aus dem Maschinenbau angewendet. Die Zusammenführung dieser unterschiedlichen Disziplinen führte zu völlig neuen Ideen und Lösungskonzepten.

Die Methoden werden jeweils prototypenhaft in Software umgesetzt, um Vorgehensweise und Nutzen an praktischen Beispielen zu erläutern. Es soll gezeigt werden, dass dadurch in nachvollziehbarer Weise möglichst frühzeitig verlässliche Aussagen über die Auswirkungen von Bauteilabweichungen getroffen werden können, und wie sich die Auswirkungen anschaulich darstellen und interpretieren lassen. Im Anschluss werden die Vor- und Nachteile der entwickelten Methoden gegenüber der bisher üblichen Vorgehensweise diskutiert.

In den beiden folgenden Unterkapiteln werden grundlegende Begriffe und Konzepte aus den Themengebieten Tolerierung und Computergrafik vorgestellt, die zum Verständnis des weiteren Texts von Bedeutung sind. Das darauffolgende Kapitel widmet sich dem Stand der Technik der Toleranzanalyse.

1.3 Tolerierung von Bauteilen

Eine technische Zeichnung bzw. ein CAD-Datensatz eines Bauteils muss, um Fertigung und Prüfung zu ermöglichen, eindeutig und vollständig sein [56]. Nach der Definition der Idealgestalt aller Bauteile muss daher für jedes funktionsrelevante Element festgelegt werden, wie weit dieses von seiner Idealgestalt abweichen darf. Diese Festlegung wird als Tolerierung bezeichnet. Erst mit der Definition von Form- und Lagetoleranzen ist ein Werkstück eindeutig und vollständig beschrieben.

1.3.1 Bauteilabweichungen

Die in der technischen Zeichnung festgelegte Nominalgestalt kann durch reale Fertigung niemals exakt erreicht werden. Es entstehen stets Abweichungen. Zur Einordnung der entstehenden Abweichungstypen lässt sich das Ordnungsschema nach DIN 4760 [23] anwenden, siehe Bild 1.

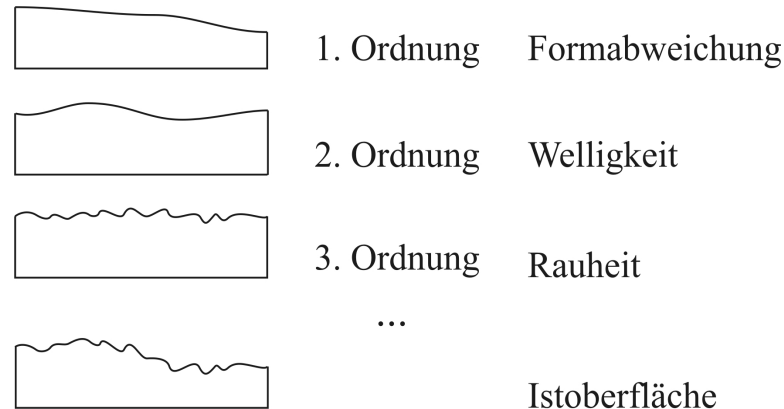


Bild 1: Überhöhter Profilschnitt unterschiedlicher Gestaltabweichungen nach DIN 4760

Die schwerwiegendste Abweichung (1. Ordnung) stellt die **Formabweichung** dar. Diese entsteht beispielsweise durch fehlerhafte Werkzeugführung, falsche Einspannung oder Durchbiegung des Werkstücks oder der Maschine, Verschleiß oder Verzug. **Welligkeit** (2. Ordnung) entsteht durch Laufabweichungen eines Fräasers oder Schwingungen bei der Fertigung. **Rauheit** (3. bis 6. Ordnung) entsteht durch abgenutzte Werkzeugschneiden, falschen Vorschub, Spanbildung oder Korrosion und äußert sich durch Rillen, Riefen und Schuppen auf der Bauteiloberfläche. Die Gestaltabweichungen überlagern sich und bilden gemeinsam die Istoberfläche, die somit nie der perfekten Nominalgestalt entspricht.

1.3.2 Toleranzvergabe in technischen Zeichnungen

Da trotz auftretender Abweichungen stets die Montierbarkeit und Funktionserfüllung jedes Bauteils gewährleistet sein muss, werden in jeder technischen Zeichnung Toleranzangaben benötigt. Diese schränken den zulässigen Bereich der Abweichungen ein. Durch **Maßtoleranzen** werden Zweipunktmessungen definiert, deren Ergebnisse in einem bestimmten Wertebereich schwanken dürfen. Maßtoleranzen sind nicht zwingend symmetrisch (z. B. 200 mm $\pm 0,1$ mm), sondern können auch asymmetrisch definiert werden (z. B. 200 mm $+0,1$ mm/ $-0,2$ mm). Erst mit Angabe von Form- und Lagetoleranzen, beziehungsweise gemeinsam mit den Tolerierungsgrundsätzen (siehe Kapitel 1.3.3), sind die Geometrielemente vollständig toleriert.

Das ISO Maßtoleranzsystem nach DIN ISO 286 [27] schränkt die Vielzahl der Tolerierungsmöglichkeiten ein und hilft, für bestimmte Funktionsanforderungen die richtigen Maßtoleranzen auszuwählen. Es vereinfacht ebenfalls die Wahl der Fertigungsmaschinen und Prüfeinrichtungen. [56]

Die ISO Maßtolerierung besteht aus dem Nennmaß, einem Kennbuchstaben und einem Zahlenwert. Der Kennbuchstabe definiert die Lage der Nulllinie, der Zahlenwert den Grundtoleranzgrad. So bedeutet beispielsweise eine Bohrung 40 H7: Die Bohrung darf minimal den Durchmesser 40 mm besitzen, maximal 40,025 mm.

In DIN ISO 1101 [28] wird das System der Form- und Lagetolerierung beschrieben. Dieses basiert auf der Definition von Lagezonen, die festlegen, in welchem räumlichen Gebiet sich das tolerierte gefertigte Geometrieelement befinden muss. Die aufgespannten Räume einzelner Form- und Lagetoleranzen sind meist Kreise oder Zylinder oder werden durch zwei parallele Geraden oder Ebenen festgelegt. Jedoch existieren auch komplexere Lagezonen.

Formtoleranzen beziehen sich immer auf ein einziges Geometrieelement eines Bauteils. Man unterscheidet zwischen realen Geometrieelementen (z. B. Kanten oder Oberflächen) und abgeleiteten Geometrieelementen (z. B. Mittelpunkten oder Achsen). Zur Überprüfung des tolerierten Elements werden die Grenzen der Lagezonen an die Istoberfläche herangeschoben, so dass der Abstand der Grenzen minimal ist. Die möglichst enge Eingrenzung wird als Tschebyschew-Kriterium bezeichnet [56]. Lässt sich das reale Formelement in Grenzen mit dem von der Toleranzdefinition angegebenen Abstand einpassen, entspricht das Element den Toleranzen, ansonsten ist das Werkstück fehlerhaft.

In DIN ISO 1101 [28] sind die Formtoleranzen Geradheit, Ebenheit, Rundheit und Zylindrizität definiert, siehe Tabelle 1.

Tabelle 1: Formtoleranzen aus DIN ISO 1101 (nach [56])




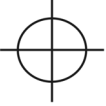



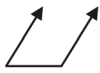


Symbol	Bezeichnung	Toleranzzone	Tolerierte Geometrieelemente
	Geradheit	Geradlinig	Alle (reale und abgeleitete) Geometrieelemente
	Ebenheit	Zwischen 2 Ebenen	
	Rundheit	Zwischen 2 Kreisen	Fast nur reale Geometrieelemente
	Zylindrizität	Ringförmig	

Lagetoleranzen besitzen neben dem tolerierten Element zusätzlich ein Bezugselement, das die Lage der Toleranzzone vorgibt. Angegeben wird das Bezugselement in der Toleranzdefinition als Großbuchstabe. Der Bezug wird über ein Bezugsdreieck und einen quadratischen Rahmen mit dem entsprechenden Großbuchstaben gekennzeichnet. Das Bezugselement ist ein reales Formelement mit Abweichungen, von dem der ideale Bezug abgeleitet wird. Durch die Definition einer Lagetoleranz wird implizit eine Formtoleranz vorgegeben, die sog. eingeschlossene Toleranz. Daher ist darauf zu achten, dass alle in einer Zeichnung angegebenen Toleranzen kleiner als die eingeschlossenen Toleranzen sind, da sie sonst keine zusätzliche Einschränkung bewirken.

Beispiel: Durch eine Tolerierung auf Parallelität vom Wert 0,3 mm wird implizit eine Tolerierung auf Geradheit von 0,3 mm generiert, da das Formelement zwischen zwei Ebenen mit Abstand 0,3 mm liegen muss.

Lagetoleranzen nach DIN ISO 1101 sind Parallelität, Rechtwinkligkeit, Neigung, Position, Koaxialität/Konzentrität, Symmetrie, Lauf und Gesamtlauf. Daneben existieren noch Profilltoleranzen, die sowohl mit als auch ohne Bezug angegeben werden können und damit nicht eindeutig den Form- oder Lagetoleranzen zugeordnet sind. Eine Übersicht gibt Tabelle 2.

Tabelle 2: Lage- und Profilltoleranzen (nach [56])

Symbol	Bezeichnung	Toleranzzone	Tolerierte Geometrieelemente
	Parallelität	Geradlinig	Alle
	Rechtwinkligkeit		
	Neigung		
	Position	Meist geradlinig, symmetrisch zum idealen Ort	Alle
	Koaxialität Konzentrität		Nur Achsen (auch der Bezug)
	Symmetrie		Meist abgeleitete
	Lauf	Ringförmig	Nur wirkliche (Bezug: nur Achsen)
	Gesamtlauf		
	Profil (einer Linie)	Beliebig, mittig zum idealen Profil	Meist wirkliche
	Profil (einer Fläche)		

In einer technischen Zeichnung werden nur die Elemente explizit toleriert, die funktionsrelevant sind und bei denen die Fertigungsgenauigkeit eine besondere Rolle spielt. Um die Gestalt der übrigen Geometrie festzulegen, werden üblicherweise **Allgemeintoleranzen** nach DIN ISO 2768 [26] angegeben. Durch Allgemeintoleranzen wird die „werkstattübliche Genauigkeit“ [56] festgelegt, die bei der Fertigung des

Bauteils erwartet wird. Allgemeintoleranzen sind verfahrensabhängig, d.h. für verbreitete Fertigungsverfahren existieren eigene Normen, die die Genauigkeit festlegen. Die Einhaltung der Allgemeintoleranzen muss nicht dauernd geprüft werden, es genügen stichprobenartige Prüfungen. Die Überschreitung von Allgemeintoleranzen führt nach DIN ISO 2768 nur dann zur Beanstandung des Werkstücks, wenn seine Funktion dadurch beeinträchtigt wird. Im Schriftfeld der Zeichnung wird die Toleranzklasse der Maß- und Winkelangaben durch einen Kleinbuchstaben (f, m, c, v für fein, mittel, grob, sehr grob) und die Form- und Lagetoleranzklasse über Großbuchstaben (H, K, L für fein, mittel, grob) angegeben. [56]

1.3.3 Tolerierungsgrundsätze

Die Tolerierungsgrundsätze definieren den Zusammenhang zwischen Maßtoleranzen und Formabweichungen, hauptsächlich bei Passungen von zylindrischen und planparallelen Passflächen [56]. Es existieren zwei Tolerierungsgrundsätze: Das Unabhängigkeitsprinzip nach DIN ISO 8015 [24] und das Hüllprinzip nach DIN 7167[25].

Das **Unabhängigkeitsprinzip** besagt: Jede Toleranz wird für sich allein geprüft. Bei alleiniger Anwendung des Unabhängigkeitsprinzips besteht die Gefahr von nicht paarungsfähigen Gleichdickformen. Bei dieser Art der Geometrieabweichung ergibt jede Messung nach dem Zweipunktverfahren einen zulässigen Wert, beispielsweise den Durchmesser d_M in Bild 2. Um jedoch eine paarungsfähige Bohrung für das Dreibogengleichdick zu erhalten, muss die Bohrung den Durchmesser d_B aufweisen.

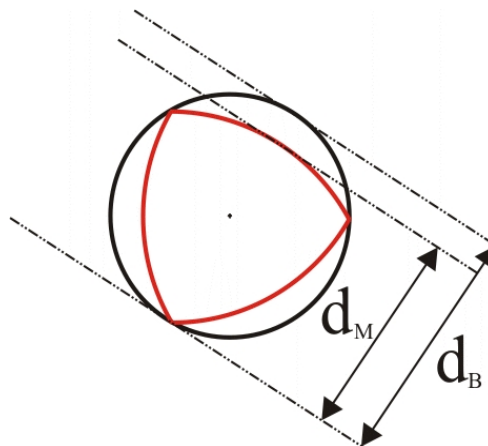


Bild 2: Dreibogengleichdick

Gleichdickformen sind nicht nur eine theoretisch mögliche Geometrieform, sondern entstehen bei gebräuchlichen Fertigungsverfahren, z. B. durch Schwingungen eines

Bohrers oder bei der Einspannung von Werkstücken in ein Dreibackenfutter. Bei parallelen Ebenen entstehen keine Gleichdickformen, hier sind Verwindungen bzw. Verdrehungen kritisch, da diese durch Zweipunktmessungen nicht nachgewiesen werden können. [56]

Um trotz Anwendung des Unabhängigkeitsprinzips die Formabweichung einzuschränken, kann bei Passungen die **Hüllbedingung** in die Zeichnung eingetragen werden. Die Hüllbedingung entspricht dem Taylorschen Prüfungsgrundsatz: “Die Gutprüfung ist eine Paarungsprüfung mit einer Lehre, die über das gesamte Geometrieelement geht.“ [56]. Bei einer Paarung Welle – Bohrung entspricht die Lehre für die Welle einem idealen Hüllzylinder mit dem Maximaldurchmesser der Welle. Durch die Hüllbedingung wird zwar die Form der Geometrieelemente eingeschränkt, nicht aber die Lage der Elemente zueinander.

Soll für alle einfachen Passelemente (Kreiszyylinder und Parallelebenenpaare) einer Zeichnung die Hüllbedingung gelten, so wird das **Hüllprinzip** nach DIN 7167 [25] angegeben. Die Hüllbedingung kann durch Einzeleintragungen durch den Konstrukteur partiell aufgehoben werden. [56], [85]

1.3.4 Auswirkungen der Toleranzvergabe

Die Tolerierung hat großen Einfluss auf die einsetzbaren Fertigungsverfahren, da nur die Verfahren geeignet sind, die eine ausreichende Genauigkeit gewährleisten. Die Tolerierung kann außerdem Nachbearbeitung von Funktionsflächen, zusätzliche Fertigungsschritte oder die Beschaffung neuer Maschinen erforderlich machen, was jeweils eine Erhöhung der Produktionskosten verursacht.

Bild 3 zeigt die Fertigungskosten einer einfachen Platte mit Bohrung in Abhängigkeit von der gewählten Toleranz (Stückzahl 12-20). Zu erkennen ist, dass die Fertigungskosten nicht stetig verlaufen, sondern sprunghaft ansteigen, wenn ein genaueres Fertigungsverfahren gewählt werden muss.

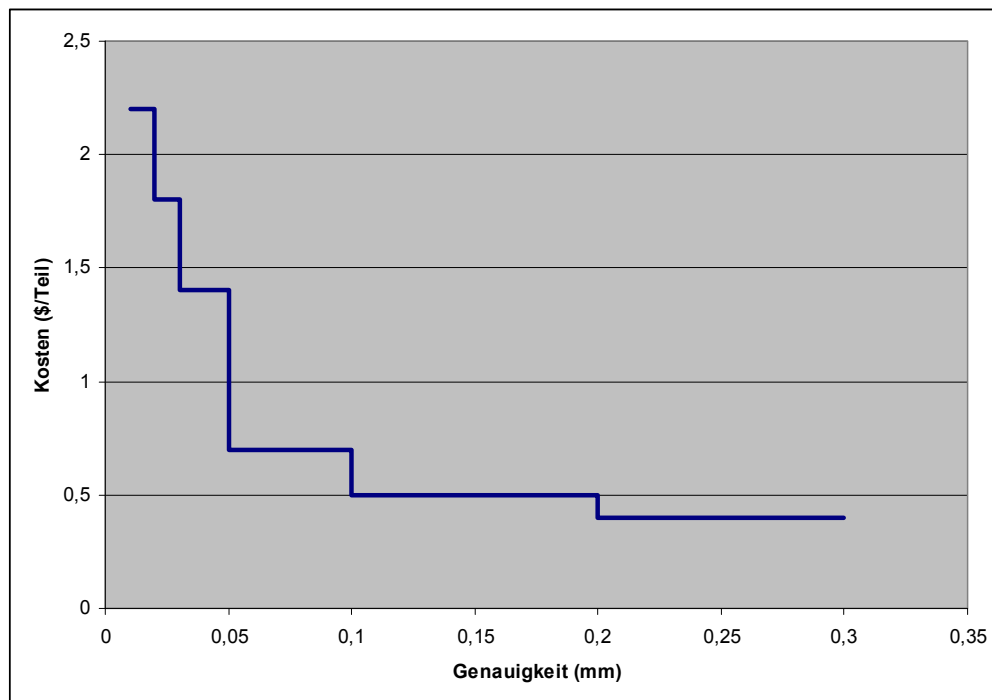


Bild 3: Herstellkosten in Abhängigkeit der Toleranz [85]

Des Weiteren beeinflusst die Toleranzvergabe die Prüfung der gefertigten Bauteile. Je nach Toleranztyp und –Wert sind unterschiedliche Messinstrumente einsetzbar, enge Toleranzen erfordern unter Umständen spezielle kosten- und zeitaufwändige Prüfeinrichtungen. Die Messunsicherheit des eingesetzten Prüfverfahrens muss außerdem miteinbezogen werden, was die Toleranzgrenzen zusätzlich einengt. Daher sollte die gewählte Toleranz immer mindestens eine Größenordnung größer als die Messunsicherheit sein [56].

Ebenfalls Einfluss hat die Toleranzsynthese auf die Montage der Bauteile zu Baugruppen. Das Zusammenfügen der Teile führt zu einer Interaktion zwischen den einzelnen Maß-, Form- und Lagetoleranzen. Treten zu große Abweichungen auf, ist die Montierbarkeit des Produkts nicht mehr gewährleistet. Man spricht von unbedingter Montierbarkeit, wenn sich Bauteile beliebig kombinieren lassen, und von bedingter Montierbarkeit, wenn die Teile zusortiert werden müssen. Unbedingte Montierbarkeit ist für die automatische Montage (z. B. durch Roboter) von großer Bedeutung.

Für den Betrieb des entstehenden Produkts haben die Toleranzen ebenfalls Auswirkungen. So werden Betriebsgeräusche, Verschleiß und Verbrauch von bewegten Teilen maßgeblich von der Toleranzvergabe bestimmt.

Neben den funktionalen Auswirkungen der Toleranzen sind in einigen Branchen (v.a. Automobilindustrie) die ästhetischen Folgen relevant, beispielsweise entstehende Spalt- und Versatzmaße der Karosserie.

Aufgrund der entstehenden Kosten lautet die Vorgabe bei der Definition von Toleranzen: „So eng wie nötig, so weit wie möglich“. Dies soll die Funktionserfüllung des Produkts über die gesamte Lebensdauer wie auch die Wirtschaftlichkeit des Produkts gewährleisten. Des Weiteren ergibt sich nach der Einzelteilmontage durch das Zusammenspiel unterschiedlicher Toleranzarten und -werte bei der Montage ein komplexes System. Dies führt dazu, dass Toleranzen oft nicht durchdacht vergeben, sondern von ähnlichen Konstruktionen übernommen werden, oder dass viel zu enge Toleranzwerte (sog. Angsttoleranzen) eingetragen werden. [56]

1.4 Grundlagen der verwendeten Computergrafik-Techniken

1.4.1 Geometrische Modellierung in CAD-Systemen

Um Geometrie am Rechner darstellen und verarbeiten zu können, muss diese durch ein geeignetes Modell abgebildet werden. Seit dem Beginn der Entwicklung von CAD-Systemen in den frühen der 1960er Jahren [124] wurden dazu verschiedenste Modelle entworfen. Die bedeutendsten Modelltypen sind Drahtmodelle, Flächenmodelle und Volumenmodelle wie Boundary Representation (B-Rep) und Constructive Solid Geometry (CSG), siehe Bild 4. In aktuellen CAD-Systemen wird meist ein Hybridsystem aus B-Rep und CSG eingesetzt. Die einzelnen Geometriemodelle werden im folgenden Text kurz vorgestellt.

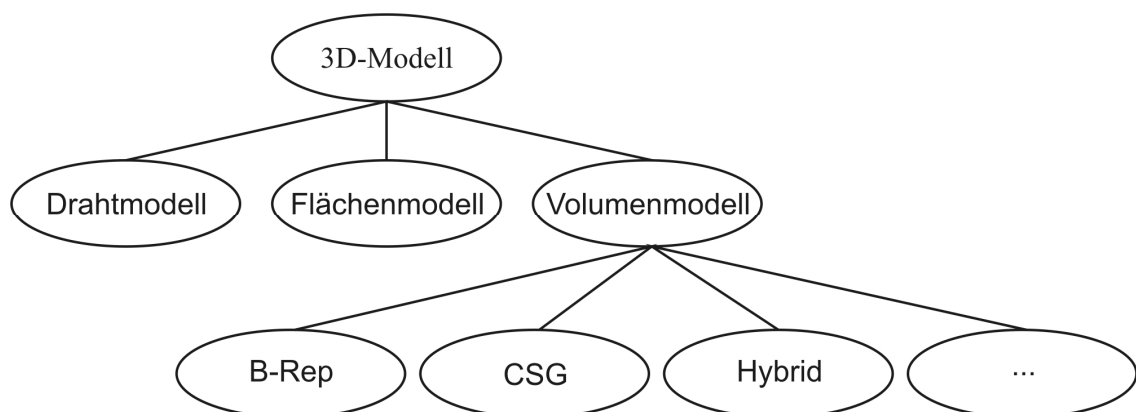


Bild 4: Überblick über 3D-Modelle nach [43]

Die ersten CAD-Systeme dienten lediglich als Ersatz für das in der Konstruktion verwendete Zeichenbrett. Sie bildeten Punkte und Linien in 2D ab, die Objekte wurden

nicht dreidimensional modelliert. Die einfachste und informationsärmste Form von 3D-Modellen sind **3D-Drahtmodelle** [43]: Es werden lediglich die Kanten des realen Objekts abgebildet, Flächen sind den Kanten nicht zugeordnet. Als Kantentypen werden meist gerade Linien, Kreise und Kreisbögen bereitgestellt. Körperinnen- bzw. Außenseite sind durch ein Drahtmodell nicht definiert. Des Weiteren ist die Darstellung als Drahtmodell in vielen Fällen mehrdeutig und bedarf der Interpretation durch den Anwender. Drahtmodelle sind daher unbrauchbar für die automatische Weiterverarbeitung, z. B. Visibilitätsermittlung, das Generieren von schattierten Darstellungen und Kollisionsberechnungen [43].

Flächenmodelle dienen zum Beschreiben von Modellen anhand der Oberflächen. Bei der Flächenmodellierung bleibt es der Sorgfalt des Anwenders überlassen, ob die modellierten Flächen einen echten dreidimensionalen Körper repräsentieren. Nur wenn das Flächenmodell das abgebildete Volumen lückenlos umschließt, ist eine Körperrepräsentation erstellt, mit der beispielsweise Volumen- und Schwerpunktberechnung möglich ist, siehe [43], S.40.

Ein Untertyp der Flächenmodelle sind Freiformflächen. Diese besitzen eine “analytisch nicht einfach beschreibbare Flächenform“ [43] mit unterschiedlichen Krümmungen in unterschiedlichen Richtungen. Anwendungsgebiete sind beispielsweise Automobil-, Flugzeug- und Schiffbau sowie der allgemeine Maschinenbau.

Freiformflächen lassen sich nicht durch Koeffizientengleichungen (wie z. B. Quadriken für Doppelkegel, Ellipsoid etc.) abbilden. Es werden interpolierende bzw. approximierende Verfahren verwendet, die die Fläche annähern. Als mathematische Beschreibung von Freiformflächen werden häufig Bézierflächen, Coons’sche Flächenpatches, Splineflächen und Nicht-Uniforme Rationale B-Splines (NURBS) verwendet, siehe [43] und [48]. Gemeinsam ist diesen Beschreibungsformen, dass sie über ein Stützpunktenetz eine begrenzte Fläche im Raum erzeugen.

Die nachfolgenden Modellierungsweisen werden als **Volumenmodellierung** [43] oder Solid Modeling [34] bezeichnet. Der Anwender bildet mit dem Volumenmodell einen realen Körper vollständig und eindeutig ab. Es ist nicht möglich, Körper mit fehlenden Elementen (Flächen/Kanten etc.) zu definieren und das Modell kann algorithmisch ohne Interpretation des Benutzers auf geometrische Fragestellungen untersucht werden. Die vom System angebotenen Erzeugungs- und Änderungsoperationen eines Volumenmodellierers führen stets wieder zu konsistenten Körpern.[43]

Die am weitesten verbreiteten Volumenmodelle sind Boundary Representation (B-Rep) und Constructive Solid Geometry (CSG).

Beim **B-Rep-Modell** wird der Körper explizit durch die ihn begrenzenden Flächenelemente beschrieben. Es wird systemintern eine verzweigte Hierarchie aufgebaut:

- Jeder Körper besteht aus begrenzenden Flächen.
- Jede Fläche besteht aus begrenzenden Kanten.
- Jede Kante trennt genau zwei Flächen und besteht aus begrenzenden Punkten.

Bild 5 zeigt beispielhaft die B-Rep Modellierung eines einfachen Körpers: Der Körper besteht aus 10 Flächen, wobei jede Fläche wiederum durch vier oder mehr Kanten beschrieben ist. Die Kanten wiederum sind über die begrenzenden Punkte definiert.

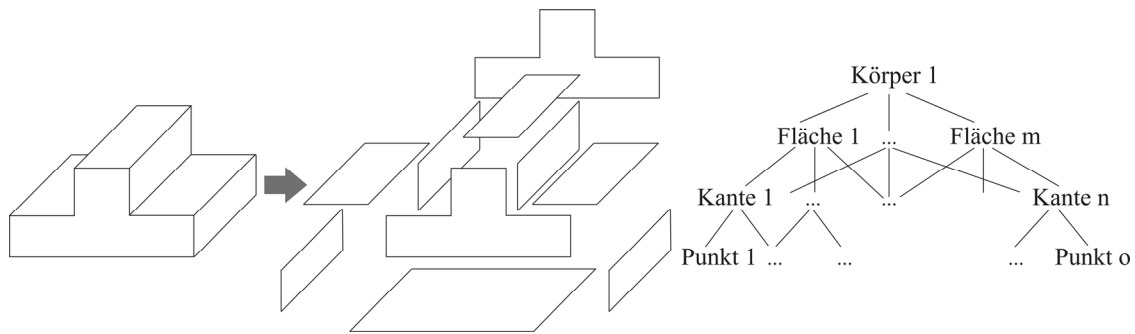


Bild 5: Modellierung eines Körpers durch Boundary Representation

Die Körperaußenseite wird entweder explizit durch eine Oberflächennormale angegeben oder ergibt sich aus der Angabereihenfolge der Flächenbegrenzungen. Im B-Rep Modell wird die Historie nicht abgelegt, d.h. es wird nicht gespeichert, durch welche Operationen die Geometrie entstanden ist, nur das Endergebnis ist enthalten. Zur graphischen Darstellung werden alle Flächen des B-Rep Modells durch Tessellierung in Dreiecksnetze (siehe Kapitel 1.4.2) umgewandelt. Da jedes Geometrieelement explizit im Datenmodell auftaucht, besteht die Möglichkeit, technologische Informationen (z. B. Oberflächengüte, Material, Toleranzen) den Geometrieelementen zuzuordnen.

Untertypen des B-Rep Modells sind das analytisch exakte Modell (erlaubt die Verwendung von Grundelementen wie Ebenen, Zylindern, Kegeln, Kugeln und Torus zur Erzeugung der Flächen), das Facettenmodell (bildet ausschließlich ebene Flächen und gerade Kanten ab) und das approximative Modell (verwendet NURBS als einzigen Flächentyp), siehe [43].

Das **CSG-Modell** unterscheidet sich grundsätzlich vom B-Rep-Modell: Es speichert nicht die Flächen des Körpers, sondern bildet diesen durch eine Folge von Operationen auf Grundkörper ab, siehe z. B. [31], [34] und [43].

Die verwendeten Grundkörper sind beispielsweise Quader, Kegel, Kugel, Zylinder. Ausgehend von den Grundkörpern lassen sich durch Anwendung von booleschen Operatoren vielfältige komplexe Körper aufbauen. Verfügbare Operatoren zur Verknüpfung der Grundkörper sind Addition, Subtraktion und Durchschnitt [34]. Gespeichert werden CSG-Modelle meist als binärer Baum, in dem die booleschen Operatoren die internen Knoten darstellen und die primitiven Grundkörper die Blattknoten [34], siehe Bild 6.

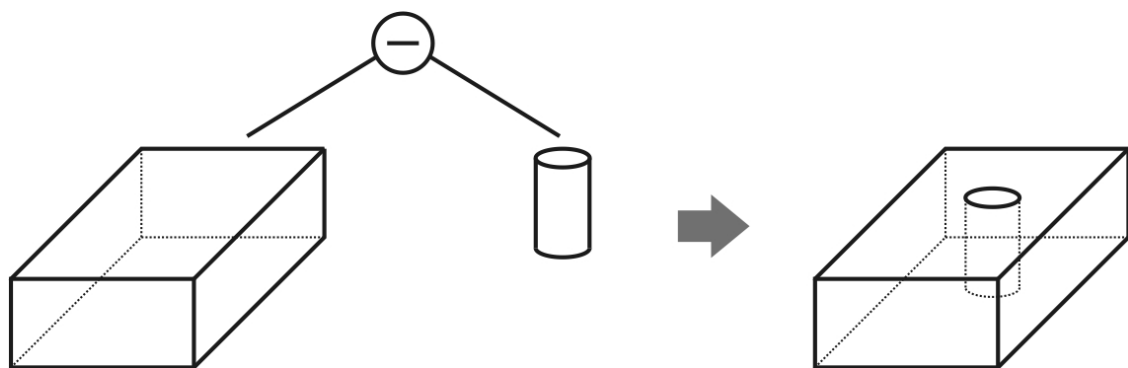


Bild 6: CSG-Modellierung einer Platte mit Bohrung durch Verknüpfung von Quader und Zylinder

Darstellbar ist ein CSG-Modell entweder über die Umwandlung in ein B-Rep-Modell oder durch Ray Tracing, wobei ein Strahl die definierten Grundkörper durchläuft und je nach verwendeter boolescher Operation die geschnittene Oberfläche dargestellt wird oder nicht.

Da beim CSG-Modell die Geometrie nicht direkt vorliegt, wird es oft als implizites [31] oder nicht evaluiertes Modell [43] bezeichnet. Dieser Umstand verhindert ein direktes Hinzufügen technologischer Information an Flächen, Kanten und Punkten [43].

Hybridmodelle sind Weiterentwicklungen des Geometriemodells, die Vorteile unterschiedlicher Repräsentationen vereinen, beispielsweise das B-Rep-Modell verwenden und zusätzlich die gesamte Entstehungshistorie der Geometrie speichern. Alternativ wird ein CSG-Modell verwendet und die B-Rep Darstellung zur Visualisierung genutzt.

1.4.2 Dreiecksnetze

Dreiecksnetze stellen eine einfache Möglichkeit dar, jegliche Geometrie zu repräsentieren. Dazu wird die Oberfläche der modellierten Geometrie durch kleine

Dreiecksfacetten angenähert. Die Facetten sind eben und besitzen drei Eckpunkte. Jeder Eckpunkt beschreibt eine Position im Raum anhand seiner x-, y- und z-Koordinaten. Weil die Oberfläche des modellierten Objekts stückweise eben approximiert wird, können gekrümmte Flächen nur angenähert, aber nicht fehlerfrei beschrieben werden, es entsteht ein Tessellierungsfehler. So ist bei grob vernetzter gekrümmter Geometrie der Fehler (z. B. an der Silhouette) deutlich zu erkennen, siehe Bild 7.

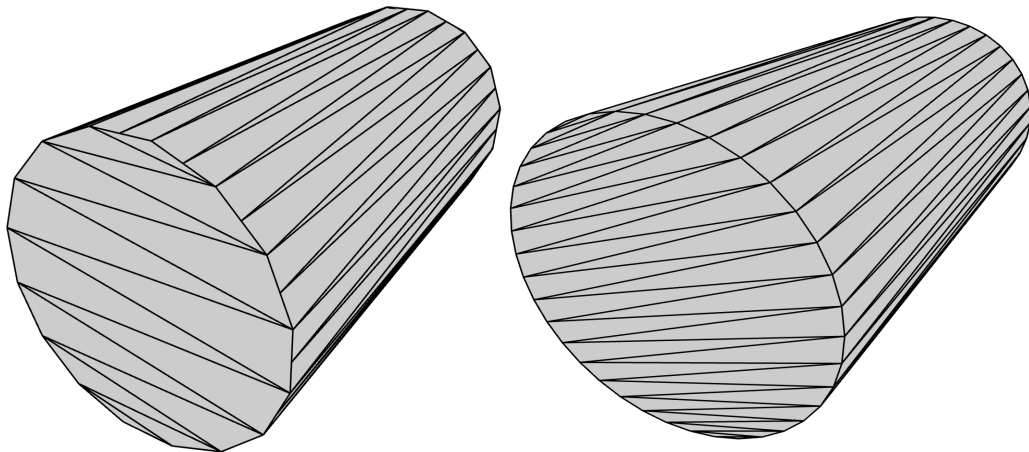


Bild 7: Vernetzung eines Zylinders mit unterschiedlichen Netzdichten (92 bzw. 148 Dreiecke)

Je nach zulässigem Tessellierungsfehler oder anderen einzuhaltenden Parametern wird eine unterschiedlich hohe Anzahl an Dreiecken zur Annäherung der Oberfläche verwendet. Der Vernetzungsgrad wirkt sich somit unmittelbar auf die Modellqualität und graphische Qualität, aber auch auf den nötigen Rechenaufwand (z. B. für Darstellung, Kollisionserkennung etc.) und Speicherbedarf (für Eckpunkte, Vernetzung, Normalen, Farbwerte etc.) aus.

Normalen sind Vektoren, die senkrecht auf der Oberfläche des Netzes stehen. Ihre Richtung zeigt üblicherweise nach „ausen“, vom beschriebenen Körper weg. Die Normalen können entweder aus einem anderen Model (z. B. Freiform-Daten) übernommen oder aus den vorliegenden Netzdaten abgeschätzt werden.

Die einfachste Möglichkeit, die Daten eines Dreiecksnetzes abzulegen ist, für jedes Dreieck die Koordinaten der drei Eckpunkte zu speichern. Dies hat den Nachteil, dass die Koordinaten bei jeder Verwendung in einem Dreieck erneut gespeichert werden [34]. Im Rahmen dieser Arbeit wurde daher das z. B. in der Virtual Reality Modelling Language (VRML, siehe [126]) spezifizierte Indexed Face Set verwendet. Hierbei werden erst die Koordinaten aller Eckpunkte angegeben und im Anschluss daran die Indices derjenigen Eckpunkte, die zusammen Dreiecke ergeben. Dies wird als Shared Vertex- Repräsentation bezeichnet. Nachteilig an der Shared Vertex-Repräsentation ist

der Aufwand, der nötig ist, um Nachbarfacetten aufzufinden: Es existieren keine Informationen über die Nachbarschaftsbeziehungen, daher wird die Repräsentation auch als „Polygon Soup“ [31] bezeichnet.

Um Nachbarfacetten schneller aufzufinden, existieren z. B. Directed Edges-Repräsentationen, bei denen die Dreiecke als Folge von gerichteten Kanten abgebildet sind, beispielsweise verwendet in [123]. Für die in dieser Arbeit nötigen Operationen war die Shared Vertex- Datenstruktur ausreichend.

Das Dreiecksnetz-Format an sich garantiert nicht, dass durch die beschriebenen Dreiecke ein sinnvoller solider Körper modelliert wird. Auftretende Fehler in Dreiecksnetz-Modellen sind etwa: Selbstüberschneidung von Dreiecken, Löcher im Netz, doppelte Eckpunkte und T-Vertices (Eckpunkte, die auf einer Kante liegen). Dies kann bei einigen Anwendungen zu Problemen und Fehlern führen, z. B. bei der Kollisionserkennung (siehe [31], Kapitel 12).

Dreiecksnetze können aus der mathematischen Geometriebeschreibung von CAD-Systemen (z. B. Bézier-Flächen oder NURBS) mittels Meshing erzeugt werden. Hierzu existieren sowohl statische Verfahren, die einmalig das gesamte Netz erzeugen, als auch dynamische Verfahren, die „on the fly“ den momentan betrachtete Szenenausschnitt vernetzen. Als Kriterium zur Beurteilung der Qualität einer Vernetzung eignen sich nach [123] folgende Parameter:

- Sehnenfehler (Maximaler Abstand zwischen Originalfläche und Netz)
- Normalenabweichung (Maximaler Winkel zwischen Flächennormale und Netznormale)
- Sehnenlänge (Länge der linearen Abschnitte)
- Sehnenwinkel (Winkel zwischen zwei aufeinanderfolgenden Flächen)

1.4.3 Kollisionsberechnung

Eine bei der Untersuchung von virtueller Geometrie sehr häufig benötigte Operation ist das Abprüfen von Kollisionen zwischen Objekten. Bei der Kollisionserkennung werden die Körper meist durch Dreiecksnetze repräsentiert. Zwar wäre die Kollisionsberechnung auch zwischen Freiformflächen möglich, allerdings ist dies erheblich aufwändiger, siehe ([73], S.75). Kollisionen von Dreiecksnetzen ließen sich in erster Herangehensweise durch paarweise Schnittberechnungen zwischen sämtlichen Dreiecken erkennen. Bei einem Brute Force Ansatz, in dem alle n Dreiecke eines Netzes mit allen n Dreiecken eines zweiten Netzes geschnitten werden, ist der

Rechenaufwand sehr hoch. Die Berechnung hat quadratische Komplexität (n^2), was bei Netzen mit sehr vielen Dreiecken zu inakzeptablen Antwortzeiten führt. Für viele Anwendungen (z. B. Spiele) reicht es aus, das ungefähre Kollisionsverhalten abzubilden. Zu diesem Zweck wird die (komplexe) Geometrie des Modells durch eine vereinfachte *Proxy Geometry* (siehe [31], S.11) mit weniger Dreiecken ersetzt. Für geometrische Toleranzanalysen ist diese Vereinfachung nicht akzeptabel.

Eine Vorgehensweise, bei komplexer Geometrie mit vielen Dreiecken die Rechenzeit zu verringern, ist es, jedes Netz in eine Hierarchie von Hüllkörpern einzufügen. Die Kollisionserkennung beginnt bei den getesteten Hierarchien damit, zwei Körper auf der obersten Hierarchieebene zu schneiden, das heißt, es werden zwei primitive Hüllkörper geschnitten, die jeweils das komplette Objekt (z. B. ein vollständiges Netz eines Bauteils) beinhalten, siehe Bild 8a. Kann festgestellt werden, dass die Hüllen sich nicht schneiden, steht bereits fest, dass auch die darunterliegenden Netze sich nicht schneiden, und die Berechnung ist für die in den Hüllen enthaltene Geometrie bereits abgeschlossen (siehe ausgegraute Bereiche in Bild 8). Schneiden sich die Hüllen, so wird der Schnitt zwischen den Hüllen auf der nächsten Hierarchieebene (Verfeinerungsstufe) einer Hülle berechnet.

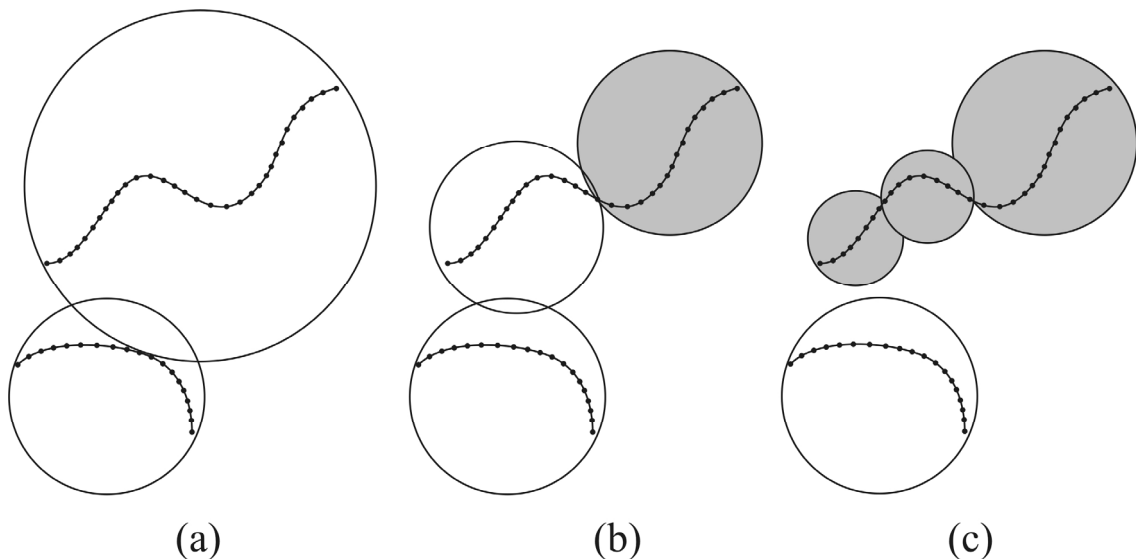


Bild 8: Kollisionserkennung unter Verwendung von Hüllkugeln

Der Algorithmus steigt die Hierarchien ab, bis entweder feststeht, dass keine Kollision vorliegt (Bild 8c), oder bis eine Kollision auf unterster Ebene (Dreieck-Dreieck) gefunden wurde. Auf diese Weise lassen sich oft große Teile der Netze aus der Kollisionsberechnung ausklammern, und die in praktischen Experimenten beobachtete

Rechenzeit sinkt stark ab. Nach dem gleichen Prinzip lassen sich auch mehrere Objekte einer Szene in einer Hülle zusammenfassen.

Als Hüllgeometrie werden meist einfache Primitive verwendet, die eine schnelle Schnittbestimmung ermöglichen, z. B. Kugeln oder an den Achsen des Koordinatensystems ausgerichtete Boxen (Axis Aligned Bounding Boxes, AABB) [31]. Um die eingehüllte Geometrie enger zu umschließen und so unnötige Schnittberechnungen einzusparen, gibt es den Ansatz, orientierte Bounding Boxen (Oriented Bounding Boxes, OBBs) zu verwenden, beispielsweise in der im Rahmen dieser Arbeit verwendeten, frei verfügbaren Kollisionsbibliothek Proximity Query Package (PQP), siehe [127]. Des Weiteren existieren Ansätze, durch *Discrete Orientation Polytopes* (k-Dops) oder konvexe Hülle die Geometrie noch enger zu umschließen [31].

PQP verwendet einen Top-Down-Ansatz, um eine Hierarchie von Hüllkörpern zu erstellen. Um eine eng anliegende Bounding Box zu erstellen, werden die Hauptachsen des Netzes bestimmt, indem die Kovarianzmatrix der Eckpunkte der konvexen Hülle des untersuchten Netzes berechnet wird. Anschließend wird die Ausdehnung der Box in alle drei Achsen bestimmt. Das beschriebene Verfahren berechnet nicht die kleinstmögliche Bounding Box, da dies mit sehr hohem Aufwand verbunden wäre, siehe [31], S. 107. Zur Erstellung der nächsten Hierarchieebene wird die errechnete Hüll-Box senkrecht zu ihrer längsten Achse in zwei Bereiche unterteilt, für die wiederum rekursiv die Ausrichtung und Hülle bestimmt wird. Die Rekursion endet, wenn die OBBs nicht weiter unterteilt werden können.

Bei Verwendung von Kugeln als Hüllgeometrie stellt sich die Schnittbestimmung einfach dar. Es muss für zwei Kugeln mit Radius $r1$ und $r2$ und Mittelpunkt $(x1,y1,z1)$ und $(x2,y2,z2)$ lediglich bestimmt werden, ob gilt:

$$(r1 + r2) < \sqrt{(x1 - x2)^2 + (y1 - y2)^2 + (z1 - z2)^2} \quad (1)$$

Falls die Summe der Radien kleiner ist als der Abstand der Kugelmittelpunkte, liegt keine Kollision vor.

Einem ähnlichen Prinzip folgt der Kollisionstest für orientierte Boxen: Um zu testen, ob sich zwei OBBs überschneiden, wendet PQP das sog. Separating Axis Theorem an, das aus dem Separating Hyperplane Theorem folgt. Dieses besagt, dass zwei konvexe Mengen A und B sich entweder schneiden, oder eine trennende Hyperebene E existiert, so dass A auf der einen Seite liegt, und B auf der anderen, siehe [31], S.156. Das

abgeleitete Separating Axis Theorem besagt, dass es ausreicht, bestimmte Achsen darauf zu prüfen, ob die Projektionen der getesteten Körper auf diese Achsen sich überlappen.

Für zwei OBBs A und B genügt es, 15 fest vorgegebene Achsen auf diese Eigenschaft zu prüfen [31], und zwar:

- Die drei Hauptachsen von A
- Die drei Hauptachsen von B
- Alle 9 Achsen, die sich aus den Kreuzprodukten der Hauptachsen von A mit den Hauptachsen von B bilden lassen

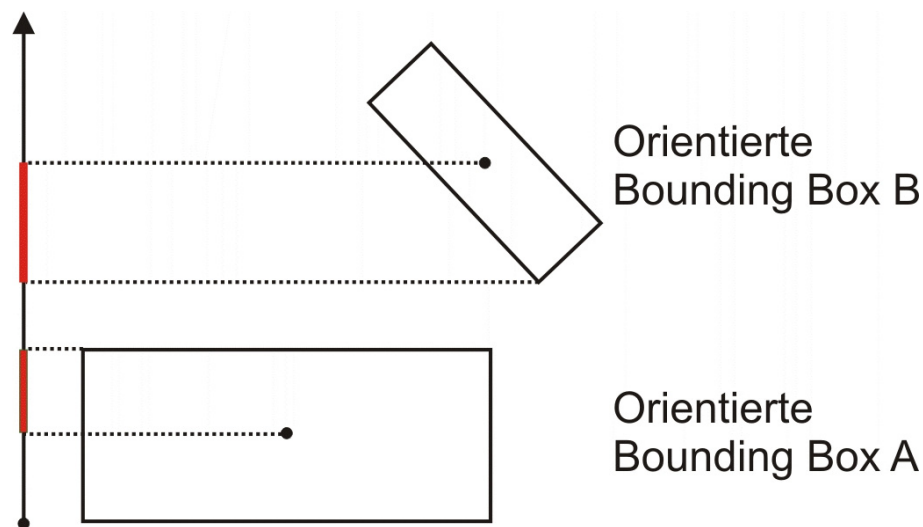


Bild 9: Separating Axis Test für eine Hauptachse von Bounding Box A

Falls die Projektionen der OBBs auf diese Achsen sich nicht überlappen, liegt eine Trennachse (Separating Axis) vor und somit existiert eine Ebene orthogonal zur Achse, die die beiden Boxen trennt. Wurde eine Trennachse gefunden, ist gezeigt, dass A und B sich nicht schneiden (siehe Bild 9). Der Umkehrschluss gilt ebenfalls: Ist keine der 15 Achsen eine Trennachse, schneiden sich die beiden Boxen. In [42] wurde gezeigt, dass selbst diese Worst-Case Berechnung (erfolgloser Test der 15 Achsen) mit weniger als 200 einfachen Rechenoperationen in sehr kurzer Zeit durchgeführt werden kann. OBBs nähern sich der umhüllten Geometrie sehr viel schneller an, als Kugeln oder AABBs, siehe Bild 10.

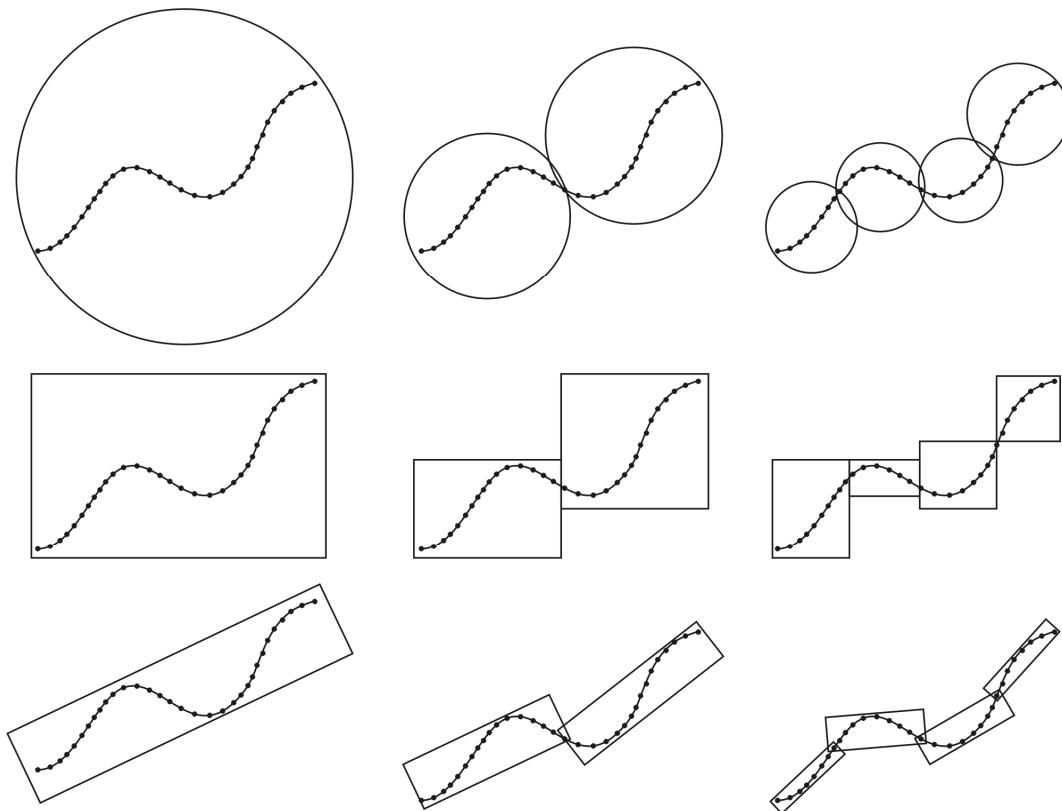


Bild 10: Einhüllen eines komplexen Körpers durch verschiedene Hülltypen (Kugeln, AABBs, OBBs)

Durch die engere Umhüllung lässt sich bei räumlich nahen Objekten ein Großteil der Schnittberechnungen einsparen. Insbesondere bei der Relativen Positionierung (Kapitel 4) und Pfadplanung (Kapitel 5) werden häufig räumlich nah zueinander liegende Objekte auf Kollision geprüft. Durch die Verwendung orientierter Boxen kann daher gegenüber einfacherer Hüllgeometrie eine erheblich bessere Antwort-Geschwindigkeit erreicht werden, obwohl der Kollisionstest der OBBs im Vergleich zu z. B. Kugeln aufwändiger ist.

1.4.4 Transformationen

Transformationen dienen dazu, Modelle im Raum zu verändern, insbesondere zu bewegen. Ein in der Computergrafik oft verwendetes Verfahren, um Transformationen zu beschreiben, sind Transformationsmatrizen, siehe z. B. [34], [2] und [128].

Durch 3x3 Matrizen lassen sich Rotationen, Scherungen und Skalierungen beschreiben. Die neuen Koordinaten jedes Modellpunkts erhält man durch Multiplikation des Punktes (Vektor aus 3 skalaren Werten) mit der Matrix:

$$p' = \begin{pmatrix} m00 & m01 & m02 \\ m10 & m11 & m12 \\ m20 & m21 & m22 \end{pmatrix} * \begin{pmatrix} px \\ py \\ pz \end{pmatrix} \quad (2)$$

Um Translationen zu beschreiben, ist eine Erweiterung dieses Konzepts nötig, die Homogene Koordinaten genannt wird. Punkte werden dabei durch einen Vektor aus 4 Komponenten beschrieben $\mathbf{p}=(px,py,pz,1)$, Vektoren als $\mathbf{v}=(vx,vy,vz,0)$. Die Matrix wird auf 4x4 Einträge erweitert, die Matriceinträge $m03$, $m13$ und $m23$ enthalten dabei die Verschiebungswerte tx , ty und tz . Durch eine Multiplikation mit der 4x4 Matrix werden Punkte korrekt rotiert und verschoben, Vektoren sind von Translationen (wie gewollt) nicht betroffen.

Sollen mehrere aufeinander folgende Transformationen auf die gleichen Daten angewendet werden, so ist es nicht nötig, die Position der Daten nach jeder Transformation schrittweise zu berechnen. Stattdessen können alle Transformationen durch Matrixmultiplikation zu einer Endmatrix zusammengefasst werden, die die gesamte Transformation beschreibt. Zu beachten ist, dass Matrixmultiplikation assoziativ ist, jedoch im Allgemeinen nicht kommutativ [2]:

$$(I \times J) \times K = I \times (J \times K)$$

$$I \times J \neq J \times I$$

Beispiel:

Ein Körper soll durch die Matrix \mathbf{M}_t zum Ursprung des Koordinatensystems verschoben werden. Hier soll der Körper durch Matrix \mathbf{M}_{rot} rotiert und anschließend durch \mathbf{M}_{inv} an die Startposition zurückverschoben werden. Die zusammengefasste Transformationsmatrix ergibt sich aus der Matrixmultiplikation

$$M_{ges} = M_{inv} \times M_{rot} \times M_t$$

Wird die Transformation von Starrkörpern beschrieben, müssen bestimmte Transformationseigenschaften eingehalten werden. Längen und Winkel müssen durch die Transformation erhalten bleiben. Es darf zu keiner Skalierung, Scherung oder Spiegelung des transformierten Körpers kommen. Diese Eigenschaften werden gewährleistet durch die spezielle Euklidische Gruppe $SE(3)$. Starrkörper besitzen 6 Freiheitsgrade, und zwar 3 translatorische und 3 rotatorische. Damit kann jeder Körper durch Angabe von 6 Parametern beliebig im Raum verschoben und verkippt werden.

Orientierungen werden oft dargestellt als eine Folge von drei Rotationen um die Grundachsen (sogenannte Eulerwinkel). Hierbei ist die Reihenfolge der Rotationen

entscheidend. Dies führt dazu, dass eine einfache Interpolation der Eulerwinkel unnatürlich wirkenden Bewegungen erzeugt, siehe [34]. Die Beschreibung von Rotationen durch Eulerwinkel hat zusätzlich den Nachteil, dass unter bestimmten Bedingungen (Zusammenfallen von Achsen durch Rotation) ein Freiheitsgrad verloren geht. Dies wird als Gimbal Lock bezeichnet [128].

Einheitsquaternionen sind wegen der genannten Nachteile von Eulerwinkeln ein bevorzugtes Verfahren, Rotationen abzubilden. Einheitsquaternionen bestehen aus einer realen Komponente und drei imaginären Komponenten i , j und k .

Quaternion $q = a + bi + cj + dk$

$$a^2 + b^2 + c^2 + d^2 = 1$$

Es gelten besondere Rechenregeln für Quaternionen:

$$i^2 = j^2 = k^2 = -1$$

$$ij = k = -ji; \quad jk = i = -kj; \quad ki = j = -ik$$

Rotationen um die Achse (b, c, d) mit den Winkel α entsprechen dem Einheitsquaternion $\cos \alpha/2 + b \sin \alpha/2 i + c \sin \alpha/2 j + d \sin \alpha/2 k$. [34]

Um Rotationen zu beschreiben, werden Quaternionen der Länge 1 benötigt. Somit sind Rotationen äquivalent zu Punkten auf der Oberfläche einer 4-dimensionalen Kugel. Gegenüber der Beschreibung von Rotationen mit Eulerwinkeln haben Quaternionen den Vorteil, dass sie besser interpoliert werden können. Um zwischen zwei Quaternionen zu interpolieren, wird die kürzeste Verbindung auf der 4D-Einheitskugel berechnet, diese Interpolation wird Sphärische Lineare Interpolation (SLERP) genannt [106].

Für zwei Quaternionen q_1 und q_2 mit $q_1 \cdot q_2 = \cos \theta$ lässt sich eine Zwischenposition u zwischen 0 und 1 folgendermaßen erzeugen:

$$slerp(q_1, q_2, u) = q_1 \cdot \frac{\sin((1-u) \cdot \theta)}{\sin \theta} + q_2 \cdot \frac{\sin(u \cdot \theta)}{\sin \theta} \quad (3)$$

Zur internen Repräsentation des räumlichen Zustands von Objekten in Programmen kann jedoch mit einer beliebigen anderen Beschreibung der Rotation (Eulerwinkel, Rotationsmatrizen) gearbeitet werden. Es reicht aus, allein die Interpolation der Zustände über die Sphärische Lineare Interpolation auszuführen, um natürlich wirkende Animationen zu erhalten.

2 Stand der Forschung und Technik (Toleranzanalyse)

Seit der Vergabe von Toleranzen auf technischen Zeichnungen um 1900 wurde eine Vielzahl von Forschungsarbeiten zur Modellierung und Analyse von Toleranzen veröffentlicht. Einen weiteren Schub bekam die Toleranzforschung, als durch den Einsatz computerbasierter Verfahren in den 1970er Jahren erste Toleranzsimulationen durchführbar wurden. In den folgenden Unterkapiteln werden die bedeutendsten Methoden der Toleranzanalyse beschrieben, geordnet nach ansteigender Modellkomplexität und ansteigendem Realitätsgrad. Neben der reinen Analysemethode wird (wenn möglich) das zugehörige Geometrie- und Toleranzmodell aufgeführt.

Einen umfassenden Überblick über Veröffentlichungen zu Modellierungsansätzen, Synthese- und Analyseverfahren für Toleranzen gibt Hong [47]. Nigam [88] beschreibt zahlreiche mathematische Analyseverfahren, Verfahren zur Behandlung von Mittelwertsverschiebungen und Probleme der statistischen Analyse von Starrkörpern. In [98] werden die theoretischen Ansätze zugeordnet, die den aufgeführten Computer Aided Tolerancing Systemen als Grundlage dienen.

In einer **Toleranzkette** hängen mehrere tolerierte geometrische Eigenschaften funktional zusammen und beeinflussen gemeinsam ein untersuchtes Endmaß, das sogenannte **Schließmaß**. Die **Schließtoleranz** bezeichnet die Breite der Schwankung des Schließmaßes, wobei der Mittelwert dieser Schwankung ggf. getrennt ermittelt werden muss.

2.1 Arithmetische Toleranzanalyse

Die grundlegendste Art der Toleranzanalyse stellt die arithmetische Analyse dar, auch Worst-Case-Analyse oder Min-Max-Analyse genannt. Diese untersucht die Auswirkungen von Einzeltoleranzen auf ein Schließmaß, indem alle Abweichungen auf den Extremwert in einer Richtung gesetzt werden. Die arithmetische Schließtoleranz T_a beschreibt die maximale Schwankungsbreite des Schließmaßes und berechnet sich aus der Summe aller in der Toleranzkette enthaltenen Einzeltoleranzen T_i .

$$T_a = \sum_{i=1}^n T_i \quad (4)$$

In diesem Kontext versteht man unter der Toleranz T_i die Differenz zwischen Höchst- und Mindestmaß. Diese besitzt kein Vorzeichen [56].

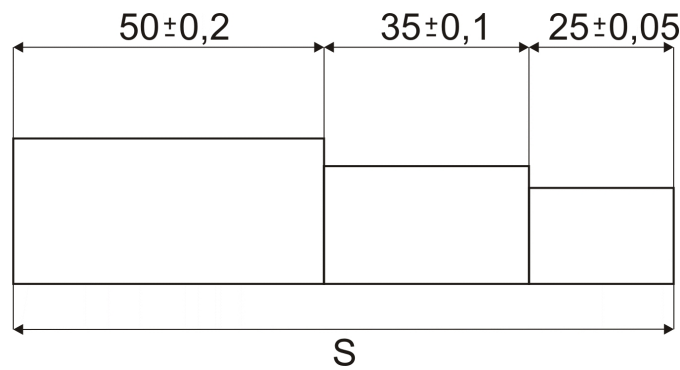


Bild 11: Einfache Toleranzkette aus Maßtoleranzen

Beispiel:

Drei einfache Bauteile werden wie in Bild 11 dargestellt zu einer Baugruppe kombiniert. Bedingung für die Funktionserfüllung der Baugruppe sei, dass die Gesamtbreite (das Schließmaß S) den Wert 110,3 mm nicht überschreitet.

Die Schließtoleranz errechnet sich zu

$$T_a = 2 \cdot 0,2 + 2 \cdot 0,1 + 2 \cdot 0,05 = 0,7 \text{ mm}$$

Da die Toleranzen jeweils symmetrisch definiert sind, liegt der Mittelwert von S bei 110 mm, der Maximalwert liegt bei 110,35 mm, der Minimalwert bei 109,65 mm. Somit kann die Funktionserfüllung der Baugruppe mit den angegebenen Toleranzen nicht garantiert werden. Eine Anpassung der Toleranzwerte ist nötig. Dies führt wie in Kapitel 1.3.4 erläutert zu einer Erhöhung der Produktkosten.

Das Auftreten dieser ungünstigsten Kombination von Extremwerten ist jedoch sehr unwahrscheinlich, da dazu jedes der drei Bauteile in seiner Maximalausprägung vorliegen muss. Betrachtet man Baugruppen aus mehreren Bauteilen, so ist damit zu rechnen, dass die einzelnen Bauteile unterschiedlich stark und in unterschiedlichen Richtungen schwanken. Daraus ergibt sich eine ausgleichende Wirkung der Schwankungen auf das Schließmaß S . Würden ausschließlich Worst-Case-Betrachtungen verwendet, wären viele aktuelle Produkte nicht herstellbar oder nicht bezahlbar (siehe [56], S.185). Um sich die ausgleichende Auswirkung zunutze zu machen, wurden zahlreiche statistische Methoden der Toleranzanalyse entwickelt, einen Überblick gibt das folgende Unterkapitel 2.2.

2.2 Statistische Toleranzanalyse

Bei der statistischen Toleranzanalyse wird davon ausgegangen, dass die betrachteten Maße innerhalb einer Toleranzkette (z. B. die Länge einer Welle) mit einer bekannten

Häufigkeitsverteilung auftreten. Oft wird als Idealisierung die Gaußsche Normalverteilung der Messwerte angenommen. Bei realen Messungen nach der Fertigung treten bei den Einzelmaßen typischerweise einseitig schiefe Verteilungen, trapezförmige Verteilungen und Nadelverteilungen auf (siehe [56]).

Der Zusammenhang zwischen Eingabeparametern und Ausgabewert von mechanischen Systemen wird meist als

$$Y = f(X_1, X_2, \dots, X_n)$$

modelliert, wobei die X_n Zufallsvariablen sind und Y als “assembly response function“ [88] bezeichnet wird. Aufgabe der statistischen Toleranzanalyse ist es, die statistischen Kenngrößen von Y abzuschätzen. Die ersten vier Momente einer Verteilung sind Mittelwert, Varianz, Schiefe und Wölbung. Wurden die Momente von Y bestimmt, lässt sich der Toleranzbereich angeben, der einen bestimmten Anteil der auftretenden Messwerte umfasst (z. B. der Bereich 64,3 mm – 64,5 mm umfasst 98% aller Ergebnisse).

2.2.1 Root Sum Squares Methode

Lässt sich Y als Linearkombination der Zufallsvariablen ausdrücken, lassen sich Mittelwert μ und Standardabweichung σ von Y aus den Kenngrößen der enthaltenen Zufallsvariablen direkt berechnen:

$$Y = a_0 + a_1 X_1 + \dots + a_n X_n$$

$$\mu_Y = a_0 + a_1 \mu_1 + a_2 \mu_2 + \dots + a_n \mu_n \quad (5)$$

$$\sigma_Y^2 = a_1^2 \sigma_1^2 + a_2^2 \sigma_2^2 + \dots + a_n^2 \sigma_n^2 \quad (6)$$

Wird davon ausgegangen, dass Y normalverteilt ist, so ist Y über diese beiden Kenngrößen vollständig definiert.

Ist Y keine Linearkombination der Zufallsvariablen, so kann f gegebenenfalls linearisiert werden. Hierfür wird beispielsweise eine Taylorreihenentwicklung von f vorgeschlagen, siehe [88].

Als pessimistische Abschätzung der zu erwartenden Schließtoleranz (bezeichnet als wahrscheinliche Schließtoleranz T_w) wurde in 2.1 die arithmetische Schließtoleranz

beschrieben, die jedoch T_w erheblich überschätzt. Zur besseren Abschätzung von T_w lässt sich als Folgerung aus Formel (6) die quadratische Schließtoleranz T_q anwenden.

$$T_q = \sqrt{\sum T_i^2} \quad (7)$$

Die quadratische Schließtoleranz bildet eine untere Grenze für T_w und ist nur erreichbar wenn gilt ([56]):

- Alle Toleranzen T_i sind voneinander unabhängig
- Alle Istabweichungen sind normalverteilt
- Die Normalverteilung liegt mittig im Toleranzfeld
- Der Prozessfähigkeitsindex C_p (siehe Anhang) ist konstant

Wenn die aufgeführten Voraussetzungen eingehalten werden, liegt die Wahrscheinlichkeit, dass T_q eingehalten wird, bei 99,73%.

In der Praxis können meist nicht alle dieser Voraussetzungen eingehalten werden, insbesondere die Unabhängigkeit und Normalverteilung der Messwerte ist meist nicht gegeben (siehe [47]) und [56]). Daher ist die wahrscheinliche Schließtoleranz größer als T_q , jedoch ist T_q meist ein erheblich besserer Schätzwert als die arithmetische Schließtoleranz T_a .

$$T_a \gg T_w \geq T_q \quad (8)$$

Beispiel:

Für das in 2.1 vorgestellte Beispiel errechnet sich T_q zu

$$\sqrt{0,4^2 + 0,2^2 + 0,1^2} = \sqrt{0,21} \approx 0,458$$

Somit liegt der Maximalwert des Schließmaßes bei 110,229 mm, der für die Funktionserfüllung zulässige Wert von 110,30 mm wird deutlich unterschritten. Fazit der statistischen Betrachtung ist, dass die Toleranzen nicht eingengt werden müssen. Es ist erkennbar, dass einzelne weite Toleranzen einen sehr negativen Einfluss auf T_q haben. Um eine kostengünstige Fertigung zu erhalten, ist es folglich ratsam, die Toleranzen einer Kette auf ähnliche Größenordnungen anzupassen. Werden alle drei Toleranzen im Beispiel auf den Wert 0,25 mm gesetzt, so ergäbe sich eine verkleinerte quadratische Schließtoleranz von 0,433 mm, obwohl die arithmetische Schließtoleranz von 0,7 mm auf 0,75 mm ansteigt.

2.2.2 Monte-Carlo-Methode

Aufgrund ihrer Einfachheit sehr weit verbreitet ist die Monte-Carlo-Simulation von Toleranzauswirkungen [88]. Hierbei werden Samples von f erzeugt, indem für jeden Eingabeparameter X_i entsprechend seiner Wahrscheinlichkeitsverteilung Werte eingesetzt werden. Die erhaltenen Ergebniswerte lassen sich im Anschluss statistisch auswerten. Vorteilhaft am Monte-Carlo-Verfahren ist, dass die Eingabeparameter beliebig verteilt sein können und f nicht analytisch vorliegen muss. Beliebige lineare und nichtlineare Response Functions sind untersuchbar. Nachteilig ist, dass für eine verlässliche Abschätzung der Momente eine sehr hohe Anzahl an Samples erzeugt werden muss, deshalb raten einige Veröffentlichungen vom Einsatz ab [88].

2.2.3 Taguchi-Methode und Erweiterungen

Eine weit verbreitete Methode zur statistischen Toleranzanalyse stammt von Taguchi [125]. Sie kann selbst dann angewendet werden, wenn die Funktion f (siehe 2.2) nicht in analytischer Form vorliegt oder die partiellen Ableitungen von f nicht gebildet werden können. Taguchis Methode basiert auf einem faktoriellen Experiment, bei dem für die Eingabeparameter X_i jeweils die drei Werte μ_i und $\mu_i \pm \sigma \cdot \sqrt{\frac{3}{2}}$ eingesetzt werden.

Nach Auswertung der 3^n möglichen Kombinationen lassen sich aus den gewonnenen Ergebnissen die statistischen Kenngrößen von f bestimmen. Die Methode bietet sich vor allem zur Auswertung kleiner Systeme mit $n < 11$ an, da hier wesentlich weniger Experimente als bei Monte-Carlo-Verfahren benötigt werden. Weiterentwickelt wurde Taguchis Methode von D'Errico und Zaino [21], um für statistische Momente 3. und 4. Grades bessere Abschätzungen zu erhalten. Dazu wird f wiederum für jeden Eingabeparameter an drei Stellen ausgewertet, jedoch mit einer angepassten Gewichtung der Werte und an angepasste Auswertestellen. In [21] wird beispielhaft aufgezeigt, dass dadurch die ersten vier statistischen Momente nach 27 Auswertungen (3 variierten Eingabeparametern) vergleichbar sind mit denen einer Monte-Carlo-Simulation nach 250.000 Auswertungen.

2.2.4 Beitragsleisteranalyse

Ziel der Beitragsleisteranalyse ist es, den Einfluss einzelner Toleranzen einer Baugruppe auf ein kritisches Schließmaß zu ermitteln und als Kenngröße anzugeben. Durch Untersuchung der Sensitivitäten kann der Produktentwickler bestimmen, welche

Toleranzen die „Stellschrauben“ zum Erreichen von bestimmten Zielmaßen sind (siehe [36]). Ermittelt werden können unterschiedliche Kenngrößen. Die einfachste Analyse stellt fest, wie sich die Änderung eines Parameters linear auf das Schließmaß auswirkt, wenn alle anderen Parameter gleich bleiben. Es wird ein konstanter Faktor berechnet, der den Funktionszusammenhang auf eine lineare Beziehung vereinfacht.

Beispiel: Die Veränderung eines Parameters um +1 mm bewirkt eine Verkleinerung des Schließmaßes um -0,32 mm, der Linearitätskoeffizient liegt folglich bei -0,32.

Ein Verfahren zur Berechnung der Linearitätskoeffizienten wird u.a. in [81] und [15] dargestellt. Um die Aussagekraft der Kenngrößen zu erhöhen, wird in der arithmetischen Beitragsleisteranalyse zusätzlich die Größe des Toleranzintervalls der Parameter in die Berechnung miteinbezogen, siehe [15] und [82]. Des Weiteren wird zur Verfeinerung der Analyse die Integration der jeweiligen Fertigungsverteilung der Parameter vorgeschlagen, siehe [82].

2.3 Modelle zur Simulation der Auswirkungen von Abweichungen

Um dreidimensional die Auswirkungen der Toleranzvergabe am Rechner zu simulieren, wurden zahlreiche Modelle entwickelt. Diese unterscheiden sich in ihrer Mächtigkeit und Detailliertheit der Problemabbildung. Die in 2.3.1 zusammengefassten Modelle abstrahieren Geometrie, Toleranzen und Zusammenbauoperationen stark. Kapitel 2.3.2 beschreibt detaillierte Modelle, die beispielsweise in der Lage sind, alle Arten von Formabweichungen korrekt abzubilden.

2.3.1 Vereinfachende Modelle zur dreidimensionalen Abweichungsbeschreibung

Um Toleranzinformationen schnell verarbeiten zu können, wurden einige vereinfachende Modelle veröffentlicht, die Auswirkungen von Bauteilvariationen (z. B. Verschiebungen und Verkippen beim Zusammenbau) berechnen. Gemeinsam ist diesen Verfahren, dass sie keine Interaktion von Starrkörpern beschreiben, sondern auf abstrakte Kontaktfeatures zurückgreifen. Diese Kontaktfeature werden z. B. als „substitute surfaces“ [9] bezeichnet.

- **Small Displacement Torsors**

Der Small Displacement Torsors (SDT) – Ansatz wurde von in [9], [8; 14] als Modell zur Berechnung der Auswirkungen von Toleranzen vorgestellt. Um die zulässige Position und Lage von Kontaktfeatures innerhalb der Toleranzgrenzen festzulegen, werden SDTs eingeführt, die Translation und Rotation der Ersatzelemente ausdrücken. Anschließend wird eine Vereinfachung der Berechnungen durchgeführt: Mit der Begründung des kleinen zulässigen Rotationsbereichs werden die Auswirkungen der Abweichungen linearisiert weiterverarbeitet. Für die Verschiebungs- und Rotationswerte werden zulässige Wertebereiche bzw. komplexere Abhängigkeiten aufgestellt. Anschließend lassen sich die aus der Toleranzdefinition resultierenden Toleranzzonen für jedes Feature berechnen.

- **Homogene Transformationsmatrizen**

Eng verwandt mit dem SDT-Ansatz ist die Matrixrepräsentation von Toleranzen. Über homogene Transformationsmatrizen (vgl. 1.4.4) werden die Versatz- und Rotationsmöglichkeiten der tolerierten geometrischen Elemente beschrieben. Ergänzt wird die Modellierung der Toleranzzone durch Bedingungen (Constraints), die die Ausdehnung der Zone begrenzen. Die Fortpflanzung von Abweichungen lässt sich folglich durch Verknüpfung der Matrizen und somit durch Matrixmultiplikationen berechnen. Beschrieben wurde der Matrix-Ansatz unter anderem für Worst-Case Betrachtungen von Achsenversatz und Verkipfung in [22]. Durch die Analyse von Extrempunkten (z. B. den Achsenendpunkten eines Zylinders) werden unter Einbeziehung der Toleranzen die möglichen Transformationen und damit die Gleichungen der Randbedingungen aufgestellt, siehe [22] und [103]. Durch Maximierung bzw. Minimierung der Gleichungen lassen sich Extremwerte des Zusammenbaus (z. B. Spaltmaße, Abstände, Verkipfungen) ermitteln. Umgesetzt wurde dieser Ansatz beispielsweise im Toleranz-Assistenzsystem FROOM, siehe [104] und [103]. Whitney et al. zeigen in [130] die Verwendung von homogenen Transformationsmatrizen zur statistischen Analyse von Toleranzauswirkungen bei Montageprozessen. Um den Wertebereich der zulässigen Variablenkombinationen (Verschiebungs- und Verkipfungswerte) abzubilden, werden die Wertebereiche zufällig gesampelt und der entstehende zulässige Werteraum z. B. durch ein Ellipsoid angenähert. Nachteil des in [130] vorgestellten Verfahrens ist, dass Formabweichungen nicht in die Berechnungen miteinbezogen werden können.

- **Topologically and technologically related surfaces (TTRS)**

Aufbauend auf dem Ansatz der Transformationsmatrizen wurde die Theorie der topologisch und technologisch zusammengehörenden Oberflächen (TTRS) präsentiert, siehe beispielsweise [17], [18] und [77].

Der TTRS-Ansatz unterscheidet sieben elementare Oberflächentypen (spärisch, planar, zylindrisch, spiralförmig, torus/kegelförmig, prismatisch und komplex). Unterscheidende Eigenschaft jedes Flächentyps ist, welche Transformationen sich ihm gegenüber invariant verhalten. Beispielsweise verhalten sich Zylinder bei Drehung um die Längsachse invariant. Eine TTRS wird in [17] definiert als eine Kopplung von zwei Oberflächen (oder einer TTRS und einer Oberfläche oder von zwei TTRS) des gleichen Festkörpers (topologischer Aspekt) in der gleichen kinematischen Schleife eines Mechanismus (technologischer Aspekt). Es wurde untersucht, welche Kopplungen technisch sinnvoll sind, so dass ungültige Paarungen (z. B. Rechtwinkligkeitstoleranz zwischen zwei parallelen Ebenen) automatisch erkannt werden [98]. Für jede TTRS wird entsprechend der definierten Toleranzen der Schwankungsbereich des jeweiligen Geometrieelements eingeschränkt. Auf diese Weise lassen sich die Auswirkungen der Toleranzdefinition auf die verschiedenen Oberflächentypen simulieren.

Neben den hier zusammengefassten Hauptrichtungen der Toleranz- und Abweichungsmodellierung existieren noch zahlreiche weitere Ansätze, beispielsweise Variationsklassen [100], Variationsräume [102] und Vektorschleifen [35; 16]. Hauptproblem aller vorgestellten Ansätze ist es, die genormte Toleranzdefinition technischer Zeichnungen nach DIN ISO 1101 [28] bzw. ANSI/ASME Y14.5M [5] semantisch exakt in eine mathematische Darstellung zu übersetzen. Ein Verzicht auf die bestehenden Standards und eine mathematische Neugestaltung der Toleranzdefinitionssprache ist nicht möglich: Einerseits müssten alle bestehenden technischen Dokumente übersetzt werden, andererseits ließe sich dadurch das in der Industrie vorhandene Erfahrungswissen zur Tolerierung nicht mehr anwenden [105]. Aus diesem Grund wird daran gearbeitet, die Tolerierungsstandards in eine mathematisch leichter handhabbare Form zu bringen.

- **T-Maps**

T-Maps wurden unter Shah und Davidson an der Arizona State University entwickelt und in einem prototypischen System umgesetzt, siehe z. B. [105], [19], [87], [6], [20] und [107]. Sie stellen eine hypothetische Punktemenge im euklidischen Raum dar, die alle zulässigen Abweichungen eines Features im Raum abbildet. Für unterschiedliche geometrische Elemente (z. B. polygonale Stäbe [87], zylindrische Elemente [19], gleichschenklige Dreiecke [20] und Langlöcher [107]) wurden T-Maps entwickelt, die mögliche Abweichungen dieser Grundtypen beschreiben. Durch die Toleranzdefinition wird die Form der T-Maps eingeschränkt. Zur Berechnung der Auswirkungen eines Zusammenbaus zweier Features lassen sich die T-Maps der gepaarten Formelemente über eine Minkowski-Summe vereinen. Durch diese Operation entsteht erneut eine T-Map. Auf diese Weise lassen sich Worst-Case Betrachtungen der kombinierten Teile erstellen. In [6] wurde gezeigt, wie sich T-Maps auch zur statistischen Toleranzanalyse nutzen lassen.

2.3.2 Bestehende geometrieorientierte Analyseverfahren

Einen grundsätzlich unterschiedlichen Ansatz zur Toleranzanalyse stellen die geometrieorientierten Verfahren („Variational Solid Models“, siehe [47]) dar. Hierbei wird die tolerierte Geometrie nicht durch stark vereinfachte Ersatzelemente repräsentiert. Es werden Instanzen der Bauteile als Geometriemodelle erzeugt, die einen hohen Detaillierungsgrad aufweisen, teilweise sogar einen geschlossenen Festkörper repräsentieren. Anhand dieser Bauteilmodelle werden die möglichen Auswirkungen der zulässigen Abweichungen überprüft. Durch die geometrieorientierte Betrachtungsweise wird das Aufstellen und Analysieren der Response Function obsolet, siehe [88]. Aus der detaillierten Abbildung und Überprüfung der Geometrie lassen sich neben den reinen Messwerten (Abstand, Versatz, Verkipfung etc.) und Versagenswahrscheinlichkeiten noch weitere Aussagen (ästhetisches Erscheinungsbild, Montierbarkeit etc.) treffen. Die Arbeitsweise zur Durchführung einer Toleranzanalyse unterscheidet sich grundlegend von den im vorherigen Kapitel vorgestellten abstrakten Verfahren. Des Weiteren wird eine Visualisierung der entstehenden nichtidealen Baugruppen ermöglicht. Die geometrieorientierte Vorgehensweise ist anschaulicher und intuitiver, fehlerhafte Modellierung lässt sich leichter aufdecken. Mögliche Toleranzprobleme lassen sich aufdecken und analysieren, ohne dass die Problemstellen bereits vor der Analyse bekannt sind. Weitere Vorteile der geometrieorientierten Analyse sind in Kapitel 3

zusammengefasst. Da die nötige Rechenleistung für eine geometrisch vollständige Bauteilrepräsentation im Vergleich zu abstrakten Verfahren ungleich höher ist, existieren erst wenige Arbeiten in diesem Zweig der Toleranzanalyse. Diese sollen im Folgenden vorgestellt werden.

- **Verfahren zur Erzeugung nichtidealer Geometriemodelle**

In [41] wird ein einfacher Ansatz vorgestellt, Modelländerungen an CAD-Daten durchzuführen. Die möglichen Änderungen sind dabei nicht auf kleine geometrische Variationen beschränkt, sondern können sogar die Topologie des Bauteils ändern. Realisiert wird dies durch ein CSG-basiertes Datenmodell, das die nachträgliche Umplatzierung von Grundkörpern über einen „relative position operator“ durchführt, der über einen einzigen skalaren Parameter definiert wird. Vorgeschlagen werden die Operatoren „parallele Verschiebung“ und „Verkipfung“. Dadurch lassen sich zwar Varianten der Bauteile erzeugen, Formänderungen (z. B. Welligkeit von Oberflächen) sind jedoch nicht möglich. Auch fehlt eine Verknüpfung von Toleranzen mit den Modellparametern.

Gupta und Turner stellen in [45] einen Ansatz vor, abweichungsbehaftete Bauteile in einem Volumenmodellierer abzubilden. Sie definieren ein Variationsmodell als geometrisches Objekt mit einer Menge an unabhängigen Variablen. Änderungen an den Variablen erzeugen geometrisch unterschiedliche Instanzen des Bauteils. Um in der Lage zu sein, Formabweichungen abzubilden, wurde nicht CSG, sondern eine B-Rep Beschreibung gewählt (siehe Kapitel 1.4.1). Die Formvariation wird realisiert durch eine Unterteilung der Bauteiloberfläche in Dreiecke. Jedem Eckpunkt wird entsprechend seiner Lage eine unterschiedliche Anzahl an Freiheitsgraden zugewiesen: Punkte innerhalb der Fläche besitzen maximal einen Freiheitsgrad (Bewegung entlang der Oberflächennormalen), Kantenpunkte maximal zwei Freiheitsgrade (falls beide angrenzenden Flächen abweichungsbehaftet sind) und Ecken bis zu drei Freiheitsgrade. Die zulässigen Wertebereiche der Modelvariablen werden durch die definierten Toleranzen bestimmt und über ein Ungleichungssystem abgebildet.

Stefano und Angelo zeigen in [112], wie sich realistische Geradheitsabweichungen von Linien generieren lassen. Dazu werden Messdaten in ein stochastisches Modell übertragen, das daraufhin in der Lage ist, NURBs-Kurven mit ähnlichen Abweichungen nachzubilden.

In [116] werden Verfahren zum Generieren nichtidealer Geometrie innerhalb von Toleranzgrenzen aufgezeigt. Schwerpunkt ist die vollständige Ausnutzung der Toleranzgrenzen. Repräsentiert werden die verformten Bauteile als Dreiecksnetze. Als Deformationsmethoden werden Free-Form Deformation (FFD) und Extended Free-Form Deformation (EFFD) beschrieben. Um nach der Deformation gültige Netze zu erhalten werden Prüfmethoden zur Ausfindung von Selbstüberschneidung und Hinterschneidungen angegeben. Zur manuellen Erstellung von Bauteilvarianten wird eine Sprache vorgeschlagen, die Raumbereichen Deformationsmethoden zuweist.

Das Forschungsprojekt VITAL (Visualizing the Impact of Tolerance Allocation, siehe [57], [58] und [83]) behandelte unter anderem das Erzeugen verformter Bauteile und Baugruppen. Zur groben Einschätzung von Spalt- und Versatzmaßen einer Baugruppe werden die einzelnen Komponenten als Starrkörper neu positioniert und orientiert. Alternativ können Spalt- und Versatzwerte an einigen Messpunkten angegeben werden und VITAL versucht, den betreffenden Starrkörper entsprechend zu transformieren. Die Ergebnisse von Toleranzanalysesimulationen (z. B. VSA) lassen sich ebenfalls nutzen: In VSA simulierte Abweichungswerte an Messpunkten werden ausgelesen und auf die Geometrie angewendet. Um die ausgelesenen Messwerte geometrisch zu erreichen, deformiert VITAL das Polygonnetz des Bauteils mit einer Kombination aus Force Density Method und Freeform Deformation (siehe [83]). Montagezwangsbedingungen und Abweichungswerte werden hierbei als Randbedingungen der Verformung behandelt. Das System hat dabei nicht den Anspruch, die Deformation physikalisch korrekt unter Berücksichtigung aller Materialkennwerte zu simulieren, sondern will nur eine grobe Abschätzung der sich einstellenden Spalt- und Versatzmaße zeigen. Eine Überprüfung, ob die so erzeugte Geometrie den Toleranzvorgaben entspricht, findet nicht statt.

- **Verfahren zur Positionierung nichtidealer Geometrie**

Werden detaillierte, nichtideale Bauteilmodelle untersucht, so ist die Simulation der Kombination von einzelnen Bauteilen zu einem Produkt ungleich schwieriger als bei den in Kapitel 2.3.1 aufgeführten Verfahren. Diese ignorieren, dass zwei gepaarte nichtideale Flächen nicht mehr perfekt zueinander passen. Bei der Simulation des Fügens von geometrisch vom Ideal abweichenden Bauteilvarianten muss jedoch bestimmt werden, welche veränderte Lage die Bauteile durch ihre Abweichungen einnehmen.

Das Registrieren der Komponenten, z. B. mit dem Iterative Closest Point (ICP, siehe [12]) Verfahren führt eine Neupositionierung eines abweichungsbehafteten Datensatzes (Data genannt) am Ideal датенatz (Model) durch, so dass der aufsummierte quadratische Abstand aller Eckpunkte von Data zu Model minimiert wird. Dies ist ausreichend, um z. B. einen Messdatensatz eines gefertigten Bauteils mit dem CAD-Modell zu vergleichen und zu erkennen, an welchen Stellen starke Abweichungen vorhanden sind. Umgebungsgeometrie wird dabei nicht beachtet.

Der Prozess der Positionierung eines abweichungsbehafteten Bauteils relativ zu einem anderen Bauteil einer Baugruppe wird als Relative Positionierung bezeichnet [111]. Im Folgenden werden unterschiedliche Ansätze der Relativen Positionierung beschrieben.

Zur Positionierung von planaren Bauteilen, die durch Verschiebung der Eckpunkte deformiert wurden, existieren zahlreiche Veröffentlichungen, z. B. [111], [90] und [76]. Bei diesen Deformationsmethoden bleiben Kanten stets gerade und Flächen stets eben, was die Problemstellung vereinfacht. Dadurch sind diese Ansätze nicht in der Lage, die für Toleranzbetrachtungen wichtigen Formabweichungen in die Positionierung mit einzubeziehen und werden nicht weiter ausgeführt.

Inui, Miura und Kimura beschreiben in [55] Relative Positionierung als Optimierungsproblem in einem rechteckig begrenzten Suchraum. Sie betrachten zweidimensionale, durch Polygone modellierte Bauteile. Somit besitzt der Suchraum drei Freiheitsgrade. Eine Konfiguration des positionierten Bauteils wird durch Angabe der Positions- und Winkeländerung ($\delta x, \delta y, \delta \theta$) angegeben. Die gesuchte Konfiguration muss zwei Bedingungen erfüllen. Sie darf nicht zu einer Kollision des positionierten Teils mit der Umgebung führen und sie muss optimal bezüglich einer angegebenen Funktion sein (z. B. Minimierung von δy). Dazu wird der Konfigurationsraum hierarchisch in Zellen unterteilt, die Zellen werden jeweils als *full* (alle Konfigurationen kollidieren), *empty* (keinerlei Kollisionen) und *mixed* klassifiziert. Die Klassifizierung geschieht mittels Schnittberechnungen des äußeren und inneren Swept Area des bewegten Bauteils mit der Umgebung. *Mixed*-Zellen werden bis zu einer angegebenen Tiefe weiter unterteilt und klassifiziert. Da nur Bewertungsfunktionen verwendet werden, die linear von δx , δy und $\delta \theta$ abhängen, lassen sich frühzeitig viele Zellen aus der Suche und Unterteilung ausschließen. Gelöst wird das Optimierungsproblem für

zweidimensionale Polygone somit über eine gezielte Suche in der aufgebauten Hierarchie.

Morière et al. verwenden in [86] das Standard Tessellation Format STL, um abweichungsbehaftete Bauteile zu beschreiben. STL speichert pro Dreieck jeweils drei Eckpunkte, so dass Punkte mehrfach gespeichert werden. Über quadratische Polynome werden die einzelnen Eckpunkte verschoben, je nach Flächenzugehörigkeit werden hierbei unterschiedlich viele Komponenten der Position verändert; es wird darauf geachtet, dass die Topologie des Netzes erhalten bleibt. Das Relative Positionierungsproblem wird wie in [55] als Optimierungsproblem behandelt, jedoch sind die Optimierungsfunktionen komplexer. Vorgeschlagen wird zum einen die Minimierung des Volumens zwischen den Oberflächen der gefügten Bauteile, alternativ (ähnlich zum ICP-Verfahren) die Minimierung der Summe aller Dreieck-Dreieck Abstände. Als Randbedingung der Minimierung wird festgelegt, dass sich die Netze nicht überschneiden dürfen. Als Optimierungsverfahren wird das Newton-Verfahren und Gradientenabstiegsverfahren verwendet, beides lokale Optimierungsverfahren.

Pierce und Rosen stellen in ihren Arbeiten (siehe [97] und [96]) Verfahren vor, um simulierte Bauteilvarianten zu analysieren. Die Varianten werden erstellt, indem reale Bauteilmessungen durch Fitting-Verfahren in NURBS-Daten überführt werden, siehe [95]. Falls keine Messungen vorliegen, werden die Fertigungsverfahren nach der aus Fachliteratur stammenden Beschreibung typischer Abweichungen nachgebildet. Relative Positionierung der Bauteilvarianten wird über lokale Optimierungsverfahren (gradientenbasierten quasi-Newton Algorithmus und Hooke-Jeeves-Verfahren) erreicht. Die in [95] beabsichtigte Implementierung einer globalen Optimierung mit Simulated Annealing wurde nicht realisiert. Als Bewertungsfunktionen werden der aufsummierte Abstand von Samplepunkten und die (höher gewichtete) Eindringtiefe der Samplepunkte in die Umgebungsgeometrie angegeben. Bewertungsfunktionen bezeichnen Funktionen, die die Güte der Lösung angeben. Wird die Bewertungsfunktion als Kostenfunktion bezeichnet, führt die Optimierung üblicherweise eine Minimierung der Kosten durch. Eine Beziehung zu realen, finanziellen Kosten muss nicht bestehen

Ziel der Analyse in [97] und [96] ist es, funktional relevante Kenngrößen der gefügten Baugruppe (z. B. Winkelversatz von Teilen) zu erhalten. Anhand eines Beispiels einer

Heftmaschine wird die Anwendung des Verfahrens beschrieben: Pro Bauteil werden wenige Varianten modelliert, z. B. eine Bauteilvariante bei hoher und eine bei niedriger Fertigungsgenauigkeit. Dadurch sind die Ergebnisse stark von einigen generierten geometrischen Bauteilvarianten abhängig. Nach der Einzelteilgenerierung werden alle möglichen Bauteilkombinationen durch die relative Positionierung erzeugt und die resultierenden Winkelabweichungen geprüft. Diese sind beim beschriebenen Heftmechanismus funktional entscheidend, da bei extremen Verkippungen die Heftklammern abknicken. Aus den Resultaten wird ermittelt, welche Oberflächen den größten Einfluss auf die Funktion besitzen und somit in hoher Genauigkeit gefertigt werden sollten. Um zusätzlich standardkonforme Toleranzangaben zu erhalten, werden die verwendeten Bauteilvarianten auf ihre Toleranzeigenschaften (z. B. Ebenheit) analysiert.

- **Bestehende Toleranzvisualisierungsverfahren**

Darstellungsmethoden zur Analyse abweichungsbehafteter Geometrie lassen sich unterteilen in möglichst realistische Darstellungen und Visualisierungsverfahren, die bestimmte Aspekte der Abweichungen besonders hervorheben.

Hervorhebende Verfahren

In [134] und den darauf aufbauenden Veröffentlichungen [65], [66] und [64] werden Verfahren zur Visualisierung von möglichen Gestaltabweichungen vorgestellt. Erzeugt werden die Bauteilvarianten der analysierten Bauteile durch zufällige Verschiebung der Eckpunkte des jeweiligen Dreiecksnetzes in Normalenrichtung. Um sicherzustellen, dass die angegebenen Formtoleranzen eingehalten werden, ist die maximale Verschiebung durch einen Skalarwert begrenzt. Da die rein zufällige Verschiebung aller Eckpunkte im Toleranzbereich sehr unrealistische Oberflächen erzeugt, wurde ein Korrekturfaktor eingeführt. Dieser beschränkt die höchste relative Abweichung zweier benachbarter Eckpunkte des Netzes. Auf diese Weise wird das Ergebnis geglättet. Die entwickelten Visualisierungsverfahren lassen sich aufteilen in Einfärbungsverfahren und geometrische Verfahren. Die nichtidealen Flächen werden als Kugelwolken (siehe Bild 12) oder skalierte Flächen angezeigt.

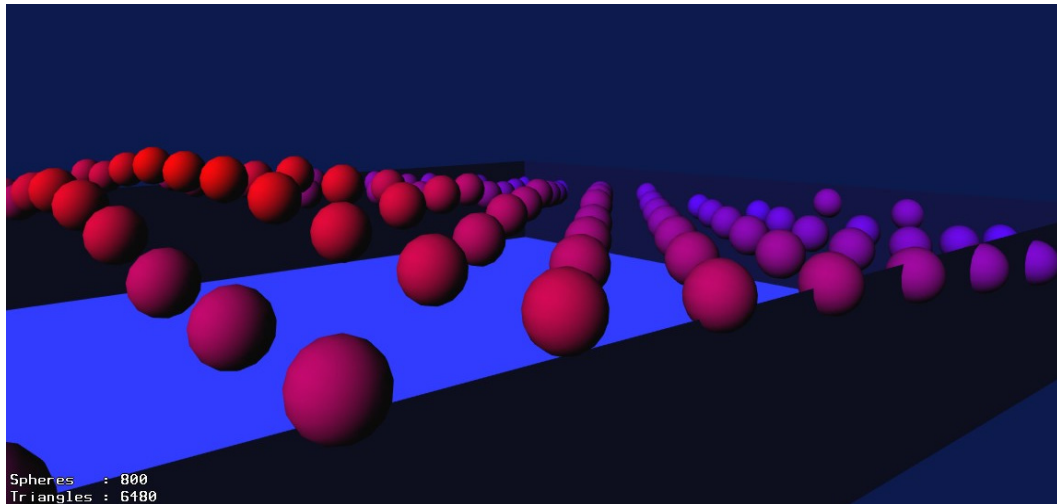


Bild 12: Visualisierung von Oberflächenabweichungen als Kugelwolke

Abweichungswerte werden über Interpolation in verschiedenen Farbräumen (Red Green Blue RGB oder Hue Saturation Value HSV) auf Farbwerte abgebildet. Alternativ werden Texturen zur Definition von Transferfunktionen verwendet, um beliebige Abbildungen von Abweichung zu Farbwert zu definieren. Beispiele für angewendete Texturen sind Einfärbungen von Abweichungswerten über bzw. unter Nominalwert mit unterschiedlichen Farben und Verwendung von Iso-Linien (siehe Bild 13), um den Höhenverlauf hervorzuheben.

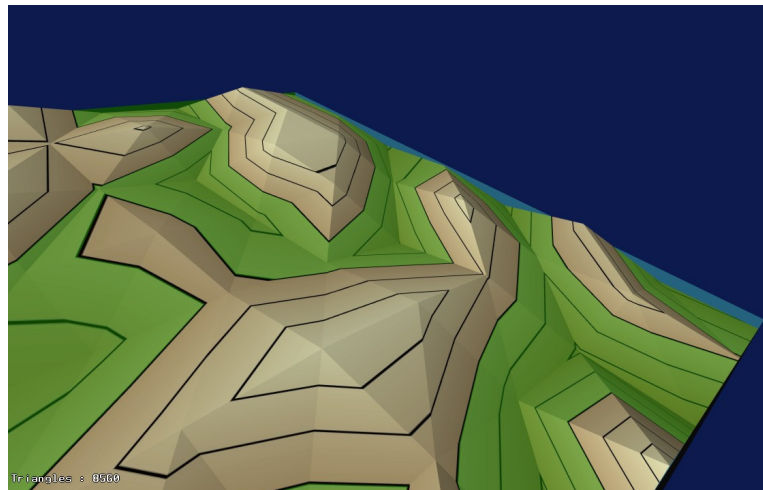


Bild 13: Hervorhebung von Oberflächenabweichungen durch Iso-Linien

Eine Farbcodierung von toleranzbedingten Abweichungen wird ebenfalls in [109] genutzt. In dem vorgestellten Tool zeigt die Farbcodierung jedoch nicht Formabweichungen, sondern die Weitergabe und Verstärkung von Abweichungen, die durch das Befestigungssystem von Blechen verursacht wird.

Lööf et al. stellen in [79] Möglichkeiten vor, Varianten des gleichen Bauteils zu einem einzigen geometrischen Objekt zu vereinen. Die gleichzeitige Darstellung aller so erstellten Objekte ergibt ein "Abweichungsvolumen", in dem alle Bauteilvarianten liegen. Das aussagekräftigste Verfahren ruft für jedes Dreieck der Bauteiloberfläche jede simulierte Position (z. B. aus einer Toleranzsimulation) ab. Die Dreiecksmenge wird durch Erstellen der konvexen Hülle zu einem dreidimensionalen Objekt vereint, das die Schwankung eines einzelnen Dreiecks abbildet. Durch Anwendung der konvexen Hülle auf jedes einzelne Dreieck (und nicht auf das Gesamtbauteil) werden z. B. Bohrungen nicht geschlossen. Verwendet wird das entstehende Abweichungsobjekt zur visuellen Abschätzung der Schwankungen und zur Kollisionsberechnung mit Umgebungsgeometrie.

In [78] wird zusätzlich vorgeschlagen, den Ansatz der konvexen Hülle zu verwenden, um gemessene Bauteilbewegungen zu visualisieren. Anhand von drei Messpunkten auf einem realen Motor werden die Transformationsmatrizen während des Betriebs rekonstruiert. Alle Eckpunkte des Dreiecksnetzes werden mit den gewonnenen Matrizen transformiert und der in [79] beschriebene Algorithmus angewendet. Als weiteres Beispiel werden die Auswirkungen von geometrischen Variationen der Haltevorrichtung auf das Aussehen einer Rückleuchte untersucht. Die analysierten Bauteile selbst liegen bei beiden Beispielen als Idealnetze ohne Formabweichungen vor.

In [117] werden Verfahren gezeigt, die es Designern ermöglichen, zulässige Spalt- und Versatzmaße festzulegen. Dies erfolgt vor der Toleranzdefinition und gibt somit die Zielwerte vor, die durch die Tolerierung erreicht werden müssen.

An mehreren Definitionspunkten entlang des Spaltverlaufs wird die Spalt- und Versatzrichtung anhand von Vektoren festgelegt, siehe Bild 14.

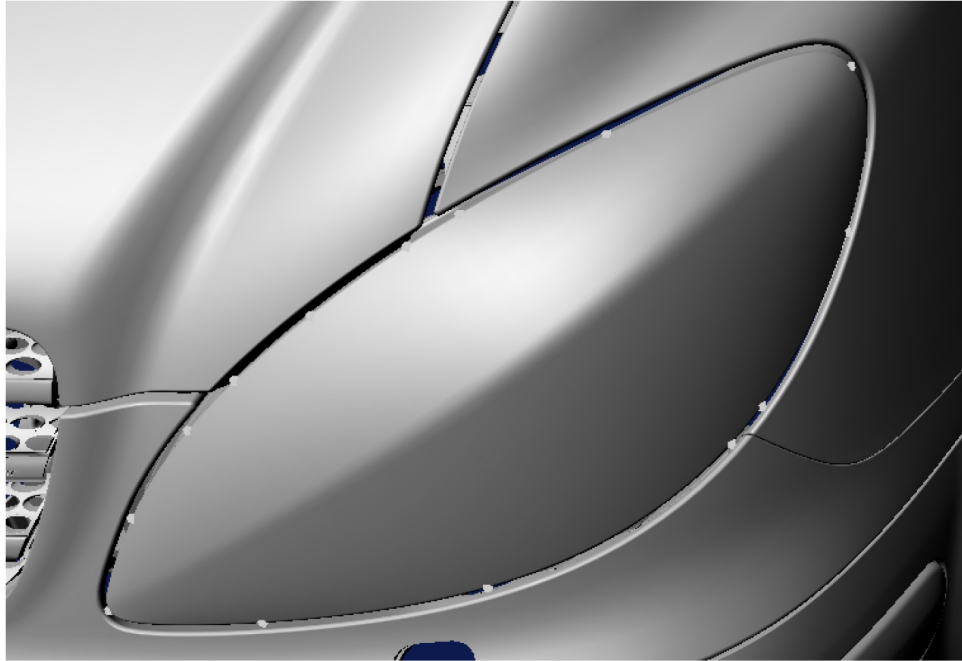


Bild 14: Definitionspunkte zur Festlegung von Spalt- und Versatzwerten [117]

Daraufhin kann der Anwender Änderungen der Spalt- und Versatzwerte zwischen Bauteilen als Änderungswert (in mm) festlegen und einstellen, welche Bauteile dazu verformt werden sollen. Ist die geänderte Lage der Definitionspunkte festgelegt, werden die Netze der Bauteile deformiert. Der Abstand der Eckpunkte des Netzes von den Definitionspunkten bestimmt den Einfluss des Verschiebevektors des jeweiligen Definitionspunkts. Um Unstetigkeiten des Netzes zu vermeiden, wird vorgeschlagen, den Abstand quadratisch oder kubisch statt linear zu berücksichtigen, siehe Bild 15.



Bild 15: Verformter Spalt mit linearem (links) und quadratischem (rechts) Einfluss des Abstands von den Definitionspunkten [117]

Um virtuelle Produktvarianten zu erstellen wird in [131] die Position von Einzelteilen geändert, indem die in CAT-Tools errechnete Platzierung des Positionierungssystems als Transformation angewendet wird. Die Form der Bauteile bleibt dabei unverändert. In [133] werden die Bauteile durch Skalierung in die jeweils gewünschte Spalt- und Versatzrichtung verändert. Als Visualisierungssoftware wird in beiden genannten Veröffentlichungen die kommerzielle Software “Opus Studio“ von Opticore eingesetzt.

In [132] untersuchen Wickman und Söderberg die Wahrnehmung von Produktvarianten in virtuellen Umgebungen. Fokus der Studie ist die Wahrnehmung von Spalt- und Versatzmaßen bei Automobilen. Für die beiden Experimente wurden unterschiedliche Einflussfaktoren ermittelt (aufgelistet in abfallender Bedeutung):

Spaltwahrnehmung ist vor allem abhängig vom Ansichtsmodus, Textur, Bauteilfarbe und Anzahl der dargestellten Umgebungsbauteile. Versatzwahrnehmung wird stark beeinflusst durch Spaltbreite, Anzahl der Umgebungsbauteile, Versatzwert und Abstand des Betrachters zu den Bauteilen.

Einen Überblick über graphisch/geometrische und statistische Toleranzanalysewerkzeuge geben Söderberg et al. in [110]. Aufgeführt sind sowohl kommerzielle Softwareprodukte als auch Forschungsprototypen, die bei Volvo im Einsatz sind. Zugeteilt sind die Werkzeuge den Phasen Konzept (Toleranzdefinition und -simulation, Robustheitsanalyse), Verifikations- und Preproductionphase (Prüfvorbereitung, Anpassung der Befestigungselemente) und Produktionsphase (Root Cause Analyse, Six Sigma Methode). Für die Konzeptphase wird vorgeschlagen, die Weitergabe und Verstärkung von Abweichungen durch Farbcodierung sichtbar zu machen. Dadurch soll eine optimale Platzierung der Befestigungselemente (z. B. Clips, Aufhängungen) von Teilen erreicht werden. Zur Überprüfung der ästhetischen Auswirkungen der Abweichungen werden Fahrzeugvarianten in einer virtuellen Umgebung realistisch abgebildet. Basierend auf den Ergebnissen der Konzeptphase lassen sich die Messpunkte der Qualitätssicherung planen, um die Gebiete mit besonders hohen erwarteten Schwankungen auf Einhaltung der definierten Qualitätskriterien zu überprüfen. Nach Angaben der Autoren konnten durch den Softwareeinsatz im Toleranzmanagement im Jahr 2003 5 Millionen Dollar bei Volvo eingespart werden.

Das bereits im Kapitel „Erzeugung nichtidealer Geometrie“ vorgestellte Forschungsprojekt VITAL setzt sich zum Ziel, die „ästhetische Qualität“ (siehe [83]) nichtidealer Bauteile (insbesondere Karosserien und das Cockpit von Automobilen) zu beurteilen. Dazu werden Fahrzeuge möglichst realistisch in einer simulierten Umgebung dargestellt. Die eingesetzten Techniken sind in [58] angegeben: Um die Umgebung bei Spiegelungen mit einzubeziehen, wird eine Cube Environment Map erstellt, die beispielsweise den Showroom zur Beurteilung des Automobils abbildet. Bi-directional Reflection Distribution Functions (BRDFs) werden verwendet, um das

Reflexionsverhalten besonderer Materialien (z. B. Metalliclackierungen) nachahmen zu können.

2.4 Umsetzung in kommerziellen Systemen

Einen Überblick über die in kommerziell verfügbaren Computer Aided Tolerancing Systemen genutzten Simulations- und Modellierungsverfahren gibt [98]. Verglichen wurden Catia.3D FDT von Dassault Systemes, CETOL 6 Sigma von Sigmetrix LLC, eM-TolMate von Tecnomatix Technologies, VSA von Engineering Animation Inc. und 3DCS von Dimensional Control Systems Inc. Von fast allen Systemen wird sowohl Worst-Case als auch statistische Analyse sowie Sensitivitätsanalyse (vgl. 2.1 und 2.2) angeboten. Zur Modellierung der Toleranzen und Abweichungen werden statistisch verteilte Punktemengen bzw. TTRS genutzt (siehe 2.3.1). Als Hauptproblem der untersuchten Systeme betrachtet [98] die automatische Verknüpfung von Toleranzvergabe, Zuordnung des Fertigungsverfahrens und Überprüfung der Funktionserfüllung von Bauteilen.

Aus dem Forschungsprojekt Vital (siehe Kapitel 2.3.2) entstand die kommerzielle Software Aesthetica, vermarktet durch Icona Solutions. Geschäftsbereiche von Aesthetica sind der Automobilbau sowie Elektrogeräte (z. B. Stereoanlagen, Küchengeräte etc.). Ziel von Aesthetica ist die schnelle Beurteilung der ästhetischen Qualität nichtidealer Konsumprodukte. Die physikalischen Auswirkungen der Toleranzvergabe lassen sich mit Aesthetica jedoch nicht vorhersagen.

Zusammenfassen lässt sich der Stand der Technik im Bereich Toleranzanalyse folgendermaßen: Es existieren bereits komplexe kommerzielle Systeme, die eine umfangreiche und zum Teil aussagekräftige Analyse vereinfachter Produktmodelle ermöglichen. Zur geometrieorientierten Toleranzanalyse existieren jedoch vergleichsweise wenige Ansätze. Die vorliegende Arbeit liefert Beiträge zur Erweiterung der Möglichkeiten dieser vielversprechenden Toleranzanalyserichtung.

3 Geometrieorientierte Toleranzanalyse: Motivation und Überblick

Im vorhergehenden Kapitel wurden die bestehenden Forschungsarbeiten und der Stand der Technik im Gebiet der Toleranzanalyse vorgestellt. Aus den dargestellten Ansätzen wird nun der Handlungsbedarf für die Weiterentwicklung der geometrieorientierten Bauteilanalyse abgeleitet.

3.1 Ableitung des Handlungsbedarfs

CAD-Systeme bilden ausschließlich die vom Konstrukteur gewünschte Idealgestalt der Bauteile ab. Die Auswirkungen von Abweichungen, die in der realen Fertigung unvermeidlich sind, werden im Computer Aided Design nicht betrachtet.

Zur Unterstützung der Toleranzspezifikation und Analyse sind zahlreiche Softwarepakete kommerziell verfügbar. Die am Markt erhältlichen Computer Aided Tolerancing-Systeme verwenden vereinfachende Geometriemodelle, die Abweichungen in die Bauteilanalyse einbringen können. Beispiele hierfür sind die in Kapitel 2.3.1 vorgestellten Ansätze wie Small Displacement Torsors, Homogene Transformationsmatrizen und T-Maps. Durch diese Ansätze werden Verschiebungen und Verkippungen von Formelementen eingebracht. Die entstehende Geometrie wird dabei jedoch nicht als vollständiges Volumenmodell erzeugt, sondern es werden die Auswirkungen auf abstrakte Ersatzelemente (z. B. Messpunkte) berechnet. Die Auswirkungen von Formänderungen der Bauteiloberflächen werden nicht betrachtet, beispielsweise bleiben Linien dadurch gerade, ebene Flächen bleiben perfekt eben. Es existiert kein CAT-System, das beliebige Geometrieabweichungen abbilden kann. Um eine Analyse komplexer Baugruppen durch Ersatzelemente durchführen zu können, ist ein sehr großer Modellierungsaufwand sowie eine hohe Abstraktionsleistung nötig. Daher wird computerunterstützte Toleranzanalyse momentan fast ausschließlich von Experten in eigenen Analyseabteilungen großer Firmen durchgeführt. Die Ergebnispräsentation der Toleranzanalyzesysteme ist ebenfalls abstrakt, meist werden Verteilungskurven von Messwerten und Beitragsleisterreports erstellt. Diese sind zwar für den Toleranzanalyseexperten aufschlussreich, eine für den verantwortlichen Konstrukteur verständliche graphische Aufbereitung der aufgedeckten geometrischen Probleme kann jedoch prinzipbedingt nicht erfolgen. Aufgrund des Abstraktionsgrads,

der hohen Systemkomplexität und der Ergebnisrepräsentation sind Fehler im Simulationsmodell sehr schwer ausfindig zu machen und bleiben oft unerkannt.

Als alternative Vorgehensweise wird die Analyse von konkreten Bauteilvarianten vorgeschlagen. Auf diesem Gebiet existieren momentan relativ wenige veröffentlichte Forschungsarbeiten verglichen mit der Vielzahl an Veröffentlichungen zur abstrakten statistischen Toleranzanalyse. Um den geometrieorientierten Ansatz für die Produktentwicklung nutzbar zu machen, sind sowohl neue Verfahren für die Geometrieerzeugung und -modellierung von Einzelteilen und Baugruppen, als auch erweiterte Methoden zur Analyse der Folgen der Abweichungen erforderlich. Die zu entwickelnden Verfahren müssen in der Lage sein, in akzeptabler Geschwindigkeit eine große Anzahl realistischer nicht-nominaler Baugruppen zu erstellen. Eine zweidimensionale oder vereinfachte Geometrirepräsentation wie in [55] und [86] reicht für einen sinnvollen Einsatz im allgemeinen Maschinenbau nicht aus. Die bloße Erzeugung von Bauteilvarianten ohne Neupositionierung innerhalb der Baugruppe (siehe [58], [83], [117] und [64]) ist zwar für bestimmte ästhetische Betrachtungen ausreichend, nicht aber für die umfassende Analyse der Auswirkungen von Abweichungen. Außerdem ist wünschenswert, dass nicht nur lokal optimale Bauteilpositionen gefunden werden ([86], [97]) sondern Verfahren eingesetzt werden, die die Bauteile verlässlich positionieren. Die entstehenden gewaltigen Datenmengen der geometrischen Simulationen müssen im Anschluss verständlich aufbereitet und analysiert werden, um statistisch gesicherte Aussagen treffen zu können. Auch hierfür sind neuartige Tools zur Messung und Visualisierung erforderlich. Der vorgeschlagene Workflow, entstehende Daten und entwickelte Softwarewerkzeuge werden in den folgenden Unterkapiteln Schritt für Schritt vorgestellt. Abschließend werden Vor- und Nachteile des Ansatzes gegenübergestellt.

3.2 Erstellung nichtidealer Bauteile

Zunächst müssen, ausgehend von einem idealen CAD-Model, geometrisch sinnvolle Varianten der einzelnen Komponenten einer Baugruppe erzeugt werden.

3.2.1 Einzelteilerzeugung

Maß-, Form und Lagetoleranzen werden im CAD-Model auf das Einzelteil aufgebracht. Nach der Toleranzdefinition können den Bauteilen Fertigungsverfahren zugewiesen werden. Im Anschluss erzeugt eine Simulation mögliche geometrische Varianten der

Bauteile. Liegen ausschließlich Toleranzen vor, so werden gültige Varianten innerhalb der Toleranzgrenzen erstellt. Selbst wenn keine Informationen zum zugeordneten Fertigungsverfahren vorliegen, ist darauf zu achten, dass realistische Bauteilvarianten entstehen. Wie in [64] und [116] beschrieben, sollten beispielsweise zwei dicht aneinander liegende Punkte auf der Bauteiloberfläche ähnliche Abweichungswerte aufweisen und nicht auf den gegenüberliegenden Grenzen des Toleranzbereichs liegen. Da durch die nichtidealen Bauteilvarianten möglichst die gesamte Toleranzzone mit einer sinnvollen Verteilung abgedeckt werden soll, ist die zufällige Variantenerzeugung ein sehr anspruchsvolles Problem, bei dem weiterhin Forschungsbedarf besteht.

Wurde den Bauteilen ein Fertigungsverfahren zugewiesen, das durch Simulation bereits hinreichend genau abgebildet werden kann, so lassen sich auf diese Weise realistische Einzelteildaten erstellen, die nahe an den Ergebnissen der tatsächlichen Fertigung liegen und die typischen Defekte des Fertigungsverfahrens widerspiegeln. Da es nicht ausreicht, nur eine mögliche Bauteilvariante zu generieren, ist es sinnvoll, Eingabeparameter der Fertigungssimulation leicht zu variieren. Beispielsweise können bei einer Tiefziehsimulation die Materialeigenschaften, Werkstückeigenschaften (z. B. Blechdicke), Prozessparameter (z. B. Niederhalterkraft) etc. innerhalb sinnvoller Grenzen entsprechend einer Versuchsmethodik frei gewählt werden, vergleiche [114] und [115].

3.2.2 Wahl des Datenformats für die Geometrieprepräsentation

Zur Modellierung der Geometrie aller Bauteile werden im vorgestellten Ansatz Dreiecksnetze verwendet. Dies hat den Vorteil, dass Eingangsdaten aus den unterschiedlichsten Systemen importiert werden können:

Für jedes verbreitete, kommerziell erhältliche CAD-System sind Vernetzer erhältlich, die das jeweilige CAD-Model in Dreiecksnetze mit vorgegebenem Vernetzungsfehler umwandeln können. Gemeinsam mit der Toleranzdefinition lassen sich automatisch oder manuell Varianten dieser Netze innerhalb des Toleranzbereichs generieren. Verfahren zur Erzeugung von Bauteilvarianten sind unter anderem in [116], [83] und [97] beschrieben.

Wie bereits erwähnt, lassen sich einige reale Fertigungsverfahren zur Herstellung von Einzelteilen bereits mit kommerzieller Software zuverlässig abbilden, beispielsweise Spritzguss (z. B. Autodesk: Moldflow, SimpaTec: Moldex3D), Gießen (z. B. Adina R & D: Adina, Magma: MagmaSoft) oder Tiefziehen (z. B. ESI: Pam-Stamp 2G, Autoform: Autoform-OneStep). Die dadurch gewonnenen Bauteilvarianten lassen sich

in Form von Dreiecksnetzen exportieren und in der virtuellen Bauteilpositionierung weiterverarbeiten. Die so erstellten nichtidealen Bauteilvarianten sind den tatsächlich entstehenden Bauteilen in Form und Abweichungsverteilung ähnlicher als zufällig oder manuell erstellte Varianten.

Zahlreiche Messsysteme tasten die Oberfläche real gefertigter Bauteile ab. Dies geschieht meist taktil (z. B. durch Koordinatenmesstechnik) oder optisch (z. B. durch Streifenprojektionsverfahren). Es entstehen durch die Messgeräte Punktwolken, die sich von der Messsoftware in Dreiecksnetze überführen lassen. So ist es möglich, gemessene Bauteile mit anderen Messdaten oder simulierten Bauteilen virtuell zu fügen. Sofern die Daten aus Volumenmodellierungssystemen (z. B. CAD-Software) stammen, lassen sich gültige Volumenmodelle exportieren, also Netze mit geschlossener Oberfläche und eindeutiger Außenseite. Für die präsentierten Methoden zur geometrieorientierten Analyse ist dies jedoch nicht zwingend erforderlich. Mit den in Kapitel 4 und 5 vorgestellten Verfahren lassen sich auch Netze verarbeiten, die aus offenen Oberflächen bestehen, beispielsweise Messdaten, bei denen manche Raumbereiche nicht erfasst wurden. Von den Dreiecken des Netzes wird keine besondere Reihenfolge oder Ordnung vorausgesetzt, die Simulationen können beliebige „Polygon Soups“ [31] verarbeiten.

Ein weiterer Grund für die Entscheidung, Dreiecksnetze als Geometrierepräsentation zu nutzen, ist die Verfügbarkeit sehr schneller Verfahren zur Detektion von Kollisionen zwischen Körpern. Prinzipiell beschrieben wurden einige dieser Verfahren bereits in Kapitel 1.4.3. Sie basieren auf der Durchsuchung einer hierarchischen Datenstruktur zum schnellen Auffinden potentieller Überschneidungen. Kollisionserkennung, die andere Repräsentationen der Geometrie verwendet (z. B. Freiformflächen) ist bedeutend aufwändiger und langsamer (siehe [31]).

Freiformflächen lassen sich durch Dreiecksnetze beliebig genau annähern. In der Praxis ist die Netzauflösung begrenzt durch die Zahlendarstellung, den Speicherbedarf und die nötige Rechenzeit für die Verarbeitung der Netze. Der Vernetzungsfehler wurde in den folgenden Experimenten so gewählt, dass er im Vergleich zu den zulässigen Geometrieabweichungen einen vernachlässigbaren Einfluss hat. Als Zahlendarstellung wurde das double-Format (64 Bit, siehe [51]) gewählt. Dies ermöglicht bei der Abbildung von Bauteilen der Größe 1000 mm einen minimalen Abstand zwischen zwei Netzpunkten von $1,11 \cdot 10^{-13} \text{ mm}$. Damit ist die Abbildung für Bauteile dieser Größenordnung hinreichend genau.

Das gewählte Datenformat zur Repräsentation von Einzelteilen kann für alle im folgenden Text vorgestellten Simulationstechniken (Relative Positionierung, Pfadplanung), Abstandsmessungen und Visualisierungsverfahren beibehalten werden. Alternativ zu Dreiecksnetzen können auch andere Geometriebeschreibungen (z. B. NURBS) verwendet werden. Allerdings müssen jeweils die notwendigen Methoden (z. B. Kollisionsberechnungen zwischen Bauteilen) implementiert werden.

3.3 Baugruppenerzeugung

Die Baugruppenerzeugung lässt sich unterteilen in die Berechnung der Endposition aller nichtidealen Bauteile (Relative Positionierung) und die Untersuchung der Montagepfades auf Kollisionsfreiheit (Pfadplanung).

3.3.1 *Relative Positionierung*

Durch die simulativ aufgebrachten Formdefekte ist es nötig, eine neue Platzierung der Bauteile innerhalb der Baugruppe zu berechnen, die Überschneidungsfreiheit garantiert und das Montageergebnis realistisch vorhersagt. Dieser als Relative Positionierung bezeichnete Prozess wurde in der vorliegenden Arbeit durch heuristische Optimierung gelöst (siehe Kapitel 4.1). Die erzeugten Einzelbauteile werden durch die Relative Positionierung gemäß ihrer Montagereihenfolge und ihres Positionierungsschemas an die Zielposition gesetzt. Jede hinzugefügte Komponente kann selbst aus mehreren (bereits positionierten) Komponenten bestehen. Hinzugefügte Bauteile werden so positioniert, dass möglichst alle vom Benutzer vorgegebenen Kriterien (z. B. Kollisionsfreiheit, Positionierungsschema) eingehalten werden und die Position bezüglich definierter Bewertungskriterien optimal ist. Schritt für Schritt kann durch das Hinzufügen weiterer nichtidealer Bauteile der Aufbau größere Baugruppen simuliert werden.

Anhand von Visualisierungsverfahren (siehe Kapitel 6) und Messverfahren können die Ergebnisse analysiert und die Einhaltung von funktionalen und ästhetischen Kenngrößen kontrolliert werden. Außerdem kann ermittelt werden, an welchen Stellen die Baugruppe in der Realität zur Qualitätssicherung geprüft werden sollte, z. B. an welchen Stellen besonders hohe, funktional relevante Abweichungen zu erwarten sind.

3.3.2 *Pfadplanung*

Durch die Pfadplanung (siehe Kapitel 5) wird ermittelt, ob es physikalisch möglich ist, die abweichungsbehafteten Bauteile in der angegebenen Reihenfolge zu montieren,

ohne dass Kollisionen auftreten. Auftretende Montageprobleme können graphisch dargestellt und statistisch ausgewertet werden. Um zu überprüfen, ob der durch Relative Positionierung ermittelte Zielzustand überhaupt erreichbar ist, muss der vom betrachteten nichtidealen Bauteil durchlaufene Weg vom Ausgangszustand (nicht montiert) zur Endposition analysiert werden. Dies wurde durch Anwendung und Anpassung von automatischen Pfadplanungsalgorithmen erreicht. Beschrieben wird die entwickelte Technik in Kapitel 5. Im Stand der Technik (Kapitel 2) wurde die Pfadplanung nicht aufgeführt, da auf diesem Gebiet keine Veröffentlichungen bekannt sind, die sich speziell der Montagesimulation toleranzbehafteter, detaillierter Geometrie widmen.

3.3.3 Monte-Carlo-Ansatz

Um eine zuverlässige Aussage über die analysierte Baugruppe treffen zu können, müssen die Schritte Einzelteilerzeugung, Relative Positionierung und Pfadplanung möglichst oft durchgeführt werden. Für Monte-Carlo-Analysen sind hierzu üblicherweise mehr als 10.000 Simulationen nötig. Daher war neben der reinen Verfahrensentwicklung die Beschleunigung der Algorithmen von entscheidender Bedeutung, da die entwickelten Verfahren wegen des Zeitaufwands sonst keinen praktischen Nutzen hätten. Ansätze hierzu waren die Parallelisierung von Berechnungen, Abschätzung von Bewertungsfunktionen und das Nutzen von Erkenntnissen aus vorherigen Berechnungen. Durch angepasste Samplingverfahren wie Latin Hypercube Sampling (siehe [115]) kann die erforderliche Anzahl der Prozessdurchläufe im Vergleich zur Monte-Carlo-Simulation bei gleichbleibender statistischer Aussagekraft der Ergebnisse reduziert werden.

3.4 Auswertungsverfahren zur Analyse abweichungsbehafteter Baugruppen

Bei Ausführung von Relativer Positionierung und Pfadplanung entstehen große Datenmengen, die geeignet ausgewertet werden müssen. Die Auswertung lässt sich unterteilen in die Analyse einzelner nichtidealer Baugruppen und in die statistische Auswertung aller aufsummierten Simulationsergebnisse.

3.4.1 Analyse einzelner Baugruppen

Bei der Betrachtung einzelner Bauteilkombinationen kann aufgezeigt werden, an welchen Stellen Montage- bzw. Positionierungsprobleme auftauchen. Die entstehenden

Bauteilkombinationen können realistisch oder durch spezielle hervorhebende Verfahren visualisiert und z. B. durch Punkt-zu-Punkt Messungen analysiert werden, siehe Kapitel 6.1

Zusätzlich zur visuellen Auswertung lässt sich der sehr hoch auflösende menschliche Tastsinn ansprechen, um abweichungsbehaftete Baugruppen zu analysieren und zu bewerten. Durch die Kopplung eines haptischen Ein-/Ausgabegeräts mit einer Echtzeit-Physiksimulation kann der Anwender Geometriedefekte mit virtuellen Prüflern erfassen. Auf diese Weise lassen sich Spalt- und Versatzwerte sowie Oberflächendefekte durch ein Haptikgerät „begreifen“. Das hierzu entwickelte prototypische Analysesystem wird in Kapitel 6.2 näher beschrieben.

3.4.2 Statistische Analyse durch Volumenvisualisierung

Um die Gesamtheit der Monte-Carlo-Ergebnisse statistisch auszuwerten, kann numerisch angegeben werden, wie bestimmte Messwerte schwanken und wie hoch der Anteil an Bauteilkombinationen ist, der nicht kollisionsfrei montierbar ist. Die numerische Auswertung der Daten kann jedoch nicht in Bezug zur Bauteilgeometrie dargestellt werden. Daher ist ein neuartiges Auswertungsverfahren wünschenswert, das zusätzlich eine graphische Analyse statistischer Ergebnisse erlaubt. Zu diesem Zweck wurde eine Volumenvisualisierung entwickelt, die alle Ergebnisse einer Monte-Carlo-Simulation in einer einzigen Visualisierung zusammenfasst. Dieses Verfahren wird in Kapitel 6.3 beschrieben.

3.5 Zusammenfassung der Geometrieorientierten Toleranzanalyse

Einen Überblick und eine thematische Einordnung der entwickelten Verfahren gibt Bild 16. Schwerpunkt der vorliegenden Dissertation stellen der Aufbau von Baugruppen aus nichtidealen Einzelteilen durch Relative Positionierung und Pfadplanung sowie die darauf folgenden Analyse- und Visualisierungstechniken dar. Nicht im Detail betrachtet wird die Thematik „Erzeugung abweichungsbehafteter Bauteilvarianten“, hier sei auf die Forschungsarbeiten [83], [97], [116] und [114] verwiesen.

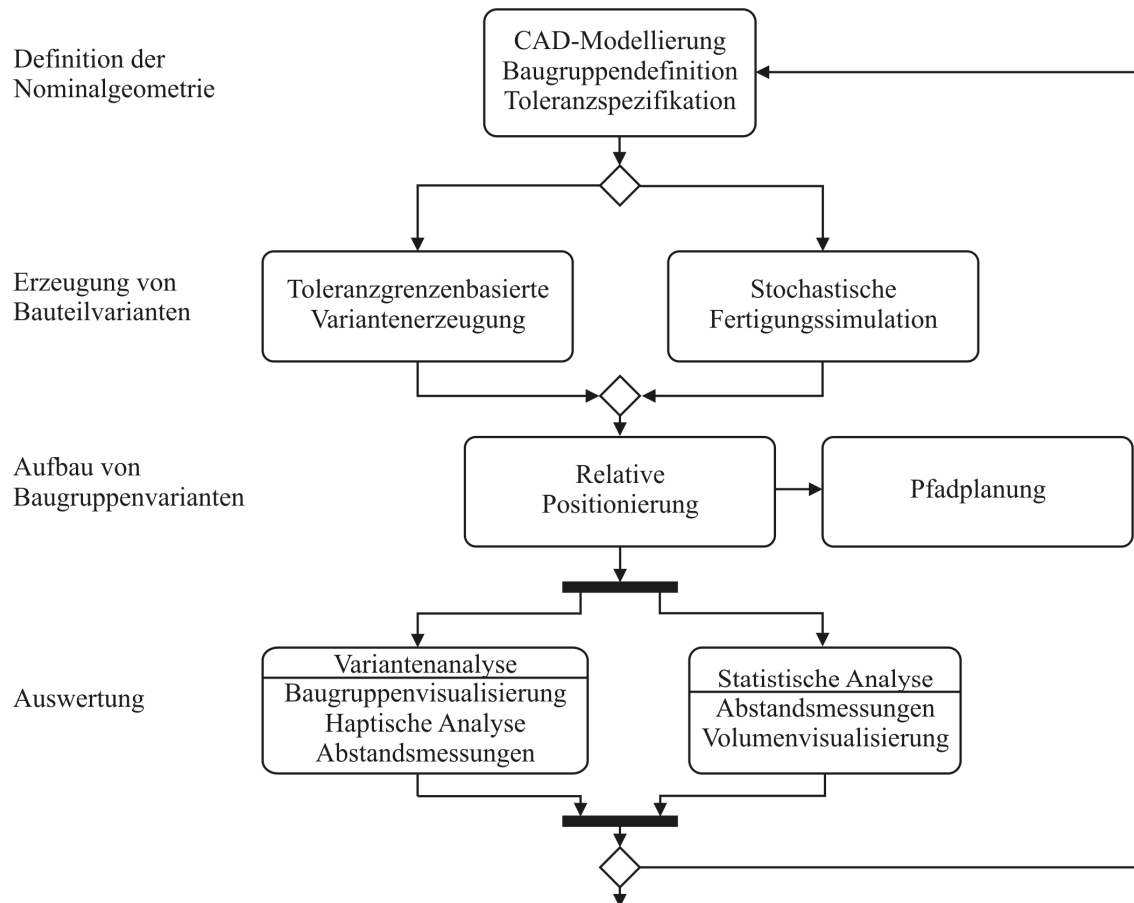


Bild 16: Übersicht der Verfahren zur geometrieorientierten Toleranzanalyse

Es ist zu beachten, dass die erzielten Ergebnisse nur dann eine gute Übereinstimmung mit der Realität besitzen, wenn die simulierten Einzelbauteile eine realistische Form und Abweichungsverteilung besitzen. Werden die Bauteile komplett zufällig innerhalb der Toleranzgrenzen erzeugt, ist dies nicht gegeben. Hauptnachteil des vorgestellten Ansatzes ist der extrem hohe Rechenaufwand im Vergleich zur abstrakten Toleranzanalyse mit Geometrieersatzelementen.

Dieser Aufwand ist gerechtfertigt, denn der Vorbereitungsprozess vor Simulationsstart ist deutlich kürzer, bleibt in der geometrischen Domäne des Produktentwicklers und ist weniger fehleranfällig als die momentan üblichen Verfahren. Durch die graphische Aufbereitung ist der Simulationsablauf besser nachvollziehbar als bei existierenden Verfahren, Ergebnisse aller Zwischenschritte können visuell überprüft werden. Es können vielfältige Eingangsdaten verwendet werden, die sowohl Maßabweichungen als auch Form- und Lageabweichungen in die Analyse integrieren. Wurden die Einzelbauteile durch stochastische Fertigungssimulationen simuliert, sind sowohl Bauteilgestalt als auch Abweichungsverteilung realistisch. Des Weiteren lassen sich die Ergebnisse der Simulationen durch umfangreichere Auswerteverfahren anschaulicher

und aussagekräftiger darstellen, als dies bisher möglich war. In aktuellen CAT-Tools sind die Messungen von Qualitätskriterien vor der Simulation zu definieren. Dies setzt voraus, dass mögliche Problemstellen bereits vor der Analyse bekannt sind, beispielsweise weil in der Montage Probleme gemeldet wurden. Werden kritische Abstände in der Simulation nicht gemessen, so lassen sich geometrische Probleme nicht aufdecken. Bei einer geometrieorientierten Analyse werden Montageprobleme stets gefunden, zusätzliche Messungen an beliebiger Stelle können auch noch im Nachhinein definiert werden.

4 Relative Positionierung

In diesem Kapitel sollen im Rahmen meiner Dissertation entwickelte Verfahren vorgestellt werden, die die Auswirkungen von Bauteilabweichungen auf die endgültige Bauteil-Lage beschreiben. Ziel der simulierten Bauteilpositionierung ist es, nichtideale Bauteile realistisch zusammenzusetzen, um zu überprüfen, welche Endgestalt Baugruppen bei vorgegebenen Toleranzen oder simulierten Fertigungsverfahren besitzen. Dadurch sollen eventuelle Probleme, beispielsweise Kollisionen, die eine Montage verhindern oder Geometrieschwankungen, die funktionale Schwierigkeiten verursachen, bereits vor der realen Fertigung aufgedeckt werden. Auf diese Weise können Änderungen am Produktmodell noch vor der Fertigung durchgeführt werden. Eine teure Nacharbeit oder Werkzeugänderung lässt sich dadurch vermeiden.

Durch die simulierte geometrische Abweichung der Einzelteile (siehe Kapitel 3.2) ist es erforderlich, für jede Bauteilvariante eine neue Position zu berechnen, da gegenüberliegende Oberflächen zweier gepaarter Bauteile keine perfekten Elemente (z. B. ebene Flächen) darstellen und folglich nicht mehr perfekt zueinander passen (siehe Bild 17). Maß-, Form- und Lageabweichungen der Bauteile machen eine Transformation des gefügten Bauteils relativ zur (fixen) Umgebung nötig. Der Positionierungsprozess wird als “Relative Positionierung” (siehe [111]) bezeichnet.

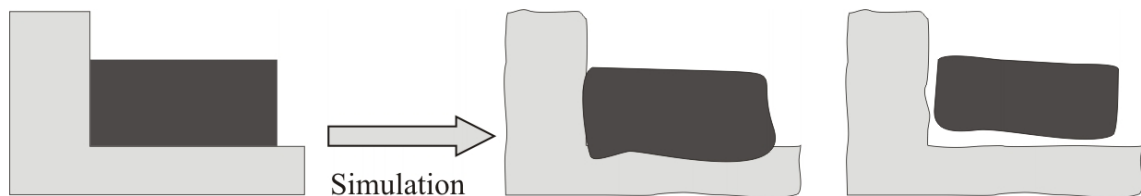


Bild 17: Nach der Simulation der Einzelteilgeometrie ist eine Neupositionierung der Teile relativ zur Umgebung notwendig

Starrkörper besitzen sechs Freiheitsgrade: Sie können in drei Raumrichtungen verschoben und um die drei Achsen rotiert werden. Eine zulässige Betrachtungsweise der Relativen Positionierung ist folglich die Bestimmung von sinnvollen Werten für diese sechs Parameter. Dazu wurde ein System entwickelt, das die Verschiebungs- und Rotationswerte über heuristische Optimierung ermittelt, siehe [118].

Die Relative Positionierung dient dazu, Schritt für Schritt Baugruppen aus nichtidealen Bauteilen zusammenzufügen. Bei jedem Positionierungsvorgang wird ein weiteres nichtideales Bauteil hinzugefügt, bis die gesamte nichtideale Baugruppe erstellt wurde.

Beeinflussen sich Teile nicht gegenseitig, können deren Positionierungsvorgänge unabhängig voneinander ausgeführt werden (siehe [119] und [120]).

4.1 Heuristische Optimierung

Um komplexe Probleme zu lösen, werden in der Informatik häufig heuristische Optimierungsverfahren eingesetzt, meist wenn eine analytische Problemlösung oder das vollständige und systematische Durchsuchen des Lösungsraums aufgrund der Größe und Komplexität des Problems nicht möglich ist. Anhand einer definierten Vorgehensweise werden Lösungen generiert und bewertet. Die beste gefundene Lösung wird vom Algorithmus zurückgegeben. Heuristische Optimierungsverfahren können oft ohne spezielle Anpassung an das jeweilige Problem verwendet werden und lassen sich somit vielseitig einsetzen.

Das Problem der Relativen Positionierung ist zu komplex, um es durch systematisches Durchsuchen des Lösungsraumes zu lösen. Zur genauen Positionierung eines Bauteils könnte beispielsweise jede Suchraumdimension in 100 diskrete Schritte unterteilt werden. Dies entspricht beispielsweise einer Schrittweite von 0,2 mm (bei einem Suchraum von ± 10 mm) bzw. $0,02^\circ$ (bei $\pm 1^\circ$) für die Positionierung eines Querträgers auf einem Automobil-Hauptboden (siehe Experiment 4.5.2). Aus der Kombination aller dadurch möglichen Positionen ergeben sich 100^6 Suchraumpositionen. Lassen sich pro Sekunde 318 Positionen bewerten, würde es 100 Jahre dauern, bis alle Kombinationen abgearbeitet sind und die beste Position bestimmt ist. Unter anderem aus diesem Grund wurde heuristische Optimierung eingesetzt, um die nichtidealen Bauteile zu positionieren. Die zur Relativen Positionierung eingesetzten Optimierungsalgorithmen orientieren sich allesamt an natürlichen Prozessen oder dem Verhalten lebender Organismen:

Simulated Annealing (SA, siehe [63]) imitiert das Abkühlverhalten von Metallen. Kühlen diese langsam genug ab, so nehmen die Atome eine nahezu ideale Position ein. Der Algorithmus betrachtet einen Punkt im Lösungsraum, der sich in jedem Zeitschritt ein wenig bewegt. Vor jeder Bewegung wird die Zielposition bewertet, fällt die Bewertung besser aus als die der momentanen Position, so wird die Bewegung durchgeführt. Fällt die Bewertung schlechter aus, wird der Schritt nur mit einer gewissen Wahrscheinlichkeit durchgeführt. Auf diese Weise kann der Algorithmus lokalen Minima entkommen. Die Akzeptanz-Wahrscheinlichkeit entspricht der

Temperatur im realen Abkühlprozess: sie ist zu Beginn der Suche hoch (nahezu jeder Schritt wird akzeptiert) und sinkt langsam ab.

Evolutionäre Algorithmen (EA) simulieren Aspekte der natürlichen Evolution und suchen nach dem Prinzip „Überleben der Bestangepassten“ (Survival of the fittest) nach einer optimalen Lösung. Einzelne Elemente im Lösungsraum werden als Individuen (Genotypen) bezeichnet, die Gesamtheit der momentan betrachteten Individuen als Population. Die Individuen werden durch die Bewertungsfunktion beurteilt, die in diesem Zusammenhang auch Fitness-Funktion genannt wird. Durch einen Auswahloperator wird bestimmt, welche Individuen überleben. Die nächste Population wird durch Anwendung von Evolutionären Operatoren erzeugt, beispielsweise Mutation (leichtes Verändern eines Individuums) und Rekombination (Mischen der Gene zweier Individuen, oft auch als Crossover bezeichnet). Der Zyklus aus Selektion, Rekombination und Mutation wird wiederholt, bis ein vorgegebenes Abbruchkriterium (z. B. Iterationszahl, Rechenzeit etc.) erreicht ist. Evolutionäre Algorithmen lassen sich in vier Hauptforschungsrichtungen einordnen: Evolutionäre Strategien ([99]), Evolutionäre Programmierung ([33]), Genetische Algorithmen ([40]) und Genetische Programmierung ([67]). Die Richtungen unterscheiden sich hinsichtlich Datenrepräsentation, Selektionsmethode und Evolutionären Operatoren.

Partikelschwarmoptimierung (PSO, siehe [60] und [61]) basiert auf der Bewegung, Intelligenz und sozialen Interaktion in Schwärmen von Lebewesen, z. B. Fisch- und Vogelschwärmen. Die einzelnen Individuen eines Schwarms werden als Partikel bezeichnet. Jedes Partikel besitzt eine Position \vec{x} im Suchraum und eine Geschwindigkeit \vec{v} . Die Partikel können untereinander kommunizieren und gute gefundene Positionen austauschen. Die Nachbarschaftstopologie bestimmt, welche Partikel miteinander kommunizieren können. Die jeweils beste Position, die ein Partikel bisher gefunden hat, speichert es als sog. kognitive Komponente \vec{p}_{best} , die in seiner Nachbarschaft bisher gefundene beste Lösung als sog. soziale Komponente \vec{g}_{best} . In jedem Zeitschritt werden Geschwindigkeit und Position für alle Partikel i im Schwarm angepasst. Die Anpassung erfolgt gemäß folgender Formel (siehe [61]):

$$\vec{v}_{i,t+1} = \omega \cdot \vec{v}_{i,t} + c_1 \cdot \vec{U}_1[0,1] \otimes (\vec{p}_{best,i,t} - \vec{x}_{i,t}) + c_2 \cdot \vec{U}_2[0,1] \otimes (\vec{g}_{best,i,t} - \vec{x}_{i,t}) \quad (9)$$

$$\vec{x}_{i,t+1} = \vec{x}_{i,t} + \vec{v}_{i,t+1}$$

Hierbei ist t der Iterationszähler, $\vec{U}_k[0,1]$ sind Zufallsvektoren mit uniform im Bereich von 0 bis 1 verteilten Komponenten und die Parameter ω , c_1 und c_2 bestimmen den Einfluss der bisherigen Position und der kognitiven bzw. sozialen Komponenten.

Das Problem der Relativen Positionierung ist im Allgemeinen nicht separabel, d.h. es ist nicht möglich, jede Dimension unabhängig zu optimieren, da eine Änderung in einer Dimension nichtlinear die Gesamtbewertung beeinflusst. Des Weiteren ist das Problem meist multimodal: es existiert nicht nur ein einziges lokales Optimum (das dadurch automatisch das globale Optimum darstellt). Vielmehr besitzt das Problem durch die nicht-nominale Gestalt der positionierten Bauteile zahlreiche lokale Optima.

Die drei vorgestellten Optimierungsverfahren beinhalten zufällige Vorgänge, beispielsweise das zufällige Verteilen der Partikel im Lösungsraum bei der Initialisierung der PSO. Daher ist die gelieferte Lösungsqualität bei mehreren Durchläufen desselben Optimierungsproblems nicht konstant, sondern schwankt, je nachdem welche Lösung gefunden wurde. Nur bei einfachen Optimierungsproblemen wird mit hoher Wahrscheinlichkeit das globale Optimum gefunden. Mitunter kann das Endergebnis eines Optimierungslaufs erheblich schlechter sein als der Durchschnittswert aller Läufe. Dies muss bei der Interpretation der experimentellen Ergebnisse beachtet werden.

4.2 Positionierungsframework

Das entwickelte Framework zur Relativen Positionierung verbindet beliebige heuristische Optimierungsverfahren über eine einheitliche Schnittstelle mit einem Positionierungsproblem. Bild 18 zeigt schematisch das Zusammenspiel der Komponenten: Der Optimierungsalgorithmus übergibt an das Problem-Model sechs Parameter (Verschiebung des Bauteils in x-, y- und z-Richtung, Rotation des Bauteils um x-, y- und z-Achse). Die übergebene Position wird von benutzerdefinierten Bewertungsfunktionen ausgewertet. Jede Bewertungsfunktion liefert für die Position eine positive Gleitkommazahl zurück, die die Qualität der Position hinsichtlich dieser Funktion ausdrückt. Schlechte Positionen werden mit hohen Werten bewertet, gute Positionen mit niedrigen Werten. Die Rückgabewerte aller Bewertungsfunktionen werden zuerst gewichtet und dann aufsummiert, die aufsummierte Bewertung wird an den Optimierer zurückgeliefert. Dieser generiert daraufhin je nach verwendeter Vorgehensweise (siehe vorhergehender Abschnitt „Heuristische Optimierung“) die

nächste sechsdimensionale Position. Nach dem Ablauf einer vorgegebenen Zahl von Iterationen liefert der Optimierer die gefundene Position mit der besten aufsummierten Bewertung (der niedrigsten Gesamtsumme) als Lösung zurück.

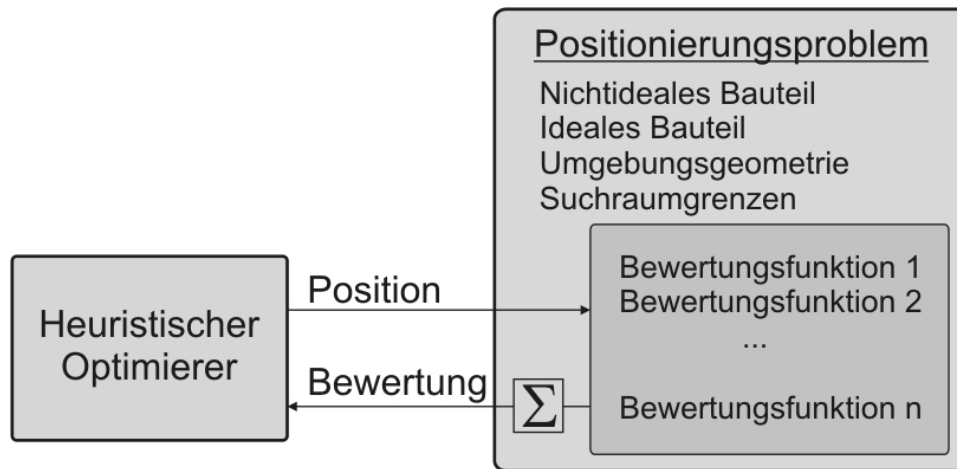


Bild 18: Framework der Relativen Positionierung [118]

Die Bewertungsfunktionen dienen dazu, eine möglichst sinnvolle und realistische Position zu finden, bei der das Bauteil beispielsweise nicht mit seiner Umgebung kollidiert, nicht schwebt und nahe an der Position des Nominalteils liegt.

Das Optimierungsframework wurde zum Zweck der Rechenzeitverringerung weiterentwickelt (siehe [39]). In [119] wurden weitere Bewertungsfunktionen und Beschleunigungsverfahren hinzugefügt. Das veröffentlichte Paper [115] beschreibt eine zusätzliche Bewertungsfunktion und die Anwendung der Relativen Positionierung auf simulierte tiefgezogene Bauteile.

4.3 Bewertungsfunktionen

Nach der in [12] definierten Nomenklatur wird das zu positionierende, ggf. nichtideale Bauteil im folgenden Text als *data* bezeichnet, das ideale Teil als *model*. Alle bereits positionierten Komponenten der Baugruppe werden als Umgebungsgeometrie bezeichnet.

4.3.1 Kollisionsvermeidung

Um Bauteilvarianten realistisch zu positionieren, ist es vor allem wichtig, dass das Dreiecksnetz von *data* nicht die Netze der Umgebungsgeometrie schneidet. Zum Sicherstellen dieser Eigenschaft wurde die frei verfügbare Kollisionsbibliothek Proximity Query Package (PQP) verwendet, siehe [127]. Die Bewertungsfunktion arbeitet binär: sie gibt bei festgestellter Überschneidung einen hohen konstanten

“Strafwert“ zurück, ansonsten Null. Die Kollisionserkennung bricht die Berechnungen ab, sobald die erste Kollision festgestellt wurde. Es wird als Parameter der Kollisionserkennung `pqp::PQP_FIRST_CONTACT` verwendet.

Das Zählen der schneidenden Dreiecke zwischen *data* und Umgebungsgeometrie wäre über den Parameter `PQP_ALL_CONTACTS` aktivierbar, wurde jedoch nicht verwendet. Eine hohe Anzahl von Überschneidungen ist nicht gleichzusetzen mit einer sehr schlechten Position: Liegt ein positioniertes Bauteil mit leichter Durchdringung auf der Umgebungsgeometrie auf, kommt es zu sehr vielen Dreieck-Dreieck-Kollisionen, siehe Bild 19, Mitte. Eine vergleichsweise schlechtere Positionierung mit tief eindringender Geometrie führt hingegen zu wenigen Kollisionen, siehe Bild 19, rechts.

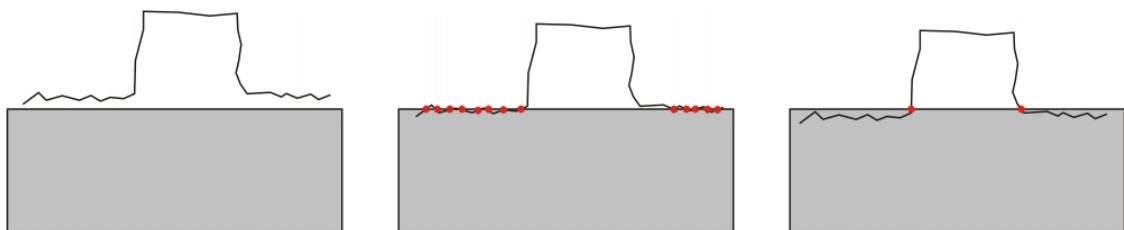


Bild 19: Kollisionen (durch rote Punkte markiert) bei der Positionierung eines nichtidealen Bauteils auf einem Quader (grau)

Weil Rechenzeit eingespart wird und die Kollisionsanzahl keine verwertbare eindeutige Qualitätseinschätzung der Position liefert, wurde die Bewertungsfunktion für Kollisionen binär realisiert.

4.3.2 Aufsummierter quadratischer Abstand (SSD)

Der aufsummierte quadratische Abstand (summed squared distance, SSD) wird häufig als Metrik zur Fehlerbewertung verwendet, beispielsweise vom Iterative Closest Point Algorithmus (ICP), siehe [12]. Für jeden Punkt des Netzes von *data* wird der euklidische Abstand zur Oberfläche von *model* berechnet. Die Abstände werden quadriert und aufsummiert.

Durch die Minimierung des SSD wird die positionierte Bauteilvariante möglichst nah an die Stelle positioniert, an der das Idealbauteil liegt. Wird keine Umgebungsgeometrie angegeben, entspricht das Ergebnis der SSD-Minimierung dem Ergebnis des ICP. Der SSD eignet sich zum Vergleich von Idealbauteil und Bauteilvariante. Ein “Schweben“ von *data* über der Umgebungsgeometrie kann durch SSD alleine nicht verhindert werden, wenn *data* kleinere Ausmaße hat als *model*, siehe Bild 17.

Die SSD-Berechnung ist für Netze mit einer hohen Anzahl an Dreiecken sehr rechenintensiv, da für jeden Eckpunkt eine Abstandsberechnung (Punkt – Netz) durchgeführt werden muss.

4.3.3 Hausdorff-Abstand

Der Hausdorff-Abstand wird in der Mustererkennung oft verwendet, um die Ähnlichkeit von zwei Objekten zu bestimmen. Der einseitige Hausdorff-Abstand ist definiert als der maximale Abstand aller Punkte von *data* zur Oberfläche von *model*. Der zweiseitige Hausdorff-Abstand beinhaltet zusätzlich eine Messung der Abstände von *model* zu *data*, wodurch auch Kerben und fehlende Bereiche von *data* in die Messung miteinbezogen werden (siehe Bild 20, rechts).

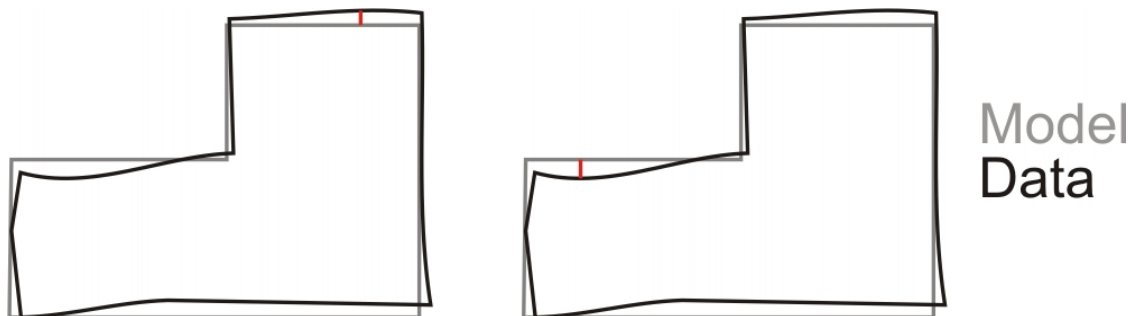


Bild 20: Hausdorff-Abstand von Data zu Model (links) und von Model zu Data (rechts)

Beim zweiseitigen Hausdorff-Abstand wird der Maximalwert beider Messungen zurückgegeben. Daher ist der nötige Rechenaufwand noch höher als bei der Berechnung des aufsummierten quadratischen Abstands: Die Abstandsberechnungen (Punkt - Netz) müssen für jeden Punkt von *model* und für jeden Punkt von *data* durchgeführt werden. Auch bei der Positionierung über diese Abstandsmetrik kann ein “Schweben“ von *data* über der Umgebungsgeometrie nicht verhindert werden, wenn *data* kleinere Ausmaße besitzt als *model*. Über die Definition zusätzlicher Bewertungsfunktionen kann dieses Problem gelöst werden.

4.3.4 Richtungsbewertung

Oft ist die Position eines Bauteils durch umgebende Bauteile oder eine Positionierungsvorrichtung bestimmt. Durch Angabe eines Richtungsvektors lässt sich in diesem Fall vorgeben, auf welche Weise *data* mit der Umgebungsgeometrie gefügt wird. Der Optimierer versucht, den Mittelpunkt von *data* (genauer: den Mittelpunkt der Axis Aligned Bounding Box um *data*) möglichst weit in die angegebene Richtung zu schieben.

Die Bewertung einer Position ergibt sich zu:

$$c = t_{\max} - \vec{v} \bullet \vec{t}$$

\vec{v} normierter Vektor, in dessen Richtung *data* verschoben wird

\vec{t} Verschiebungsvektor der aktuellen 6D-Position

t_{\max} maximal mögliche Verschiebung des Mittelpunkts in Richtung \vec{v} innerhalb des Suchraums

- Skalarprodukt

Durch \vec{v} wird festgelegt, welche Verschiebungsrichtung eine höhere Bedeutung hat, siehe Bild 21. Auf diese Weise kann näherungsweise die in der Realität oft verwendete 3-2-1 Positionierung nachgebildet werden, die beispielsweise folgendermaßen definiert ist: Das Bauteil wird zunächst auf eine Ebene aufgelegt, dadurch verliert es zwei rotatorische Freiheitsgrade und einen translatorischen Freiheitsgrad. Daraufhin wird es an eine Kante geschoben und verliert einen weiteren rotatorischen und einen translatorischen Freiheitsgrad. Durch einen letzten, punktförmigen Anschlag wird der verbleibende (translatorische) Freiheitsgrad fixiert und die Bauteilposition ist vollständig definiert.

Der Optimierer bestimmt bei Minimierung der Richtungsbewertung einen Kompromiss der Verschiebungswerte. Kann z. B. ein viel besserer Wert in x-Richtung erreicht werden, während sich die y-Komponente dadurch nur leicht verschlechtert, so wird diese Position (je nach Richtung von \vec{v}) besser bewertet.

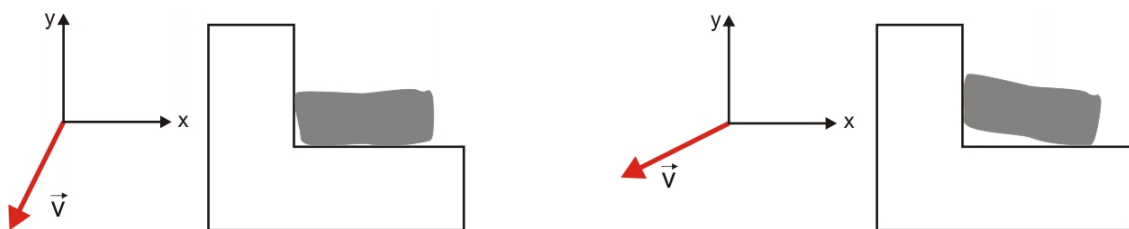


Bild 21: Unterschiedliche Positionierung eines Bauteils bei höherer Gewichtung der -y-Richtung (links) bzw. höherer Gewichtung der -x-Richtung (rechts)

Durch eine Relative Positionierung anhand der Bewertungsfunktionen Kollisionsvermeidung und Richtungsbewertung kann ein “Schweben“ des Bauteils vermieden werden, indem der angegebene Richtungsvektor in die entsprechende Richtung der Auflagefläche des positionierten Bauteils zeigt.

Da die Richtungsbewertung nur aus dem Bilden eines Skalarprodukts und einer Subtraktion besteht, lässt sie sich im Vergleich zu den bereits beschriebenen

Bewertungsfunktionen (z. B. Kollisionsvermeidung) mit sehr geringem Rechenaufwand auswerten.

4.4 Verfahren zur Beschleunigung der Positionierung

Da für die Durchführung einer Relativen Positionierung eine große Anzahl von Aufrufen der (zum Teil komplexen) Bewertungsfunktionen nötig ist, kann der Vorgang sehr viel Zeit in Anspruch nehmen. Aus diesem Grund wurden (gemeinsam mit A. Gnezdilov, S. Helwig, S. Stoll und G. Kókai) verschiedene Ansätze zur Beschleunigung der Relativen Positionierung implementiert, siehe [118] und [39]. Die Ansätze werden in den Unterkapiteln 4.4.1 bis 4.4.3 vorgestellt.

4.4.1 Vorzeitiger Berechnungsabbruch beim Aufsummieren

Wie Formel (9) zeigt, wird bei der Partikelschwarmoptimierung pro Partikel nur die Bewertung der jeweils besten bisher gefundenen Position als kognitive Komponente gespeichert. Die Berechnung der Gesamtbewertung einer Position besteht aus der Bildung einer Summe aus den Rückgabewerten der einzelnen Bewertungsfunktionen. Die Bewertungsfunktionen liefern stets positive Werte. Daher kann der Berechnungsvorgang vorzeitig abgebrochen werden, sobald die momentan ermittelte Summe größer (und damit schlechter) als die kognitive Komponente des aktuellen Partikels ist. Die Gesamtbewertung liefert in diesem Fall INFINITY (die größtmögliche Double-Zahl) zurück. Das Schwarmverhalten wird durch diesen Abbruch nicht beeinflusst, die durchschnittliche Gesamtrechnzeit der Relativen Positionierung konnte bei den durchgeführten Tests auf bis zu ein Drittel reduziert werden (siehe [118]).

4.4.2 Abschätzen der Bewertung durch Auswahl weniger Vertices

Um Berechnungen zu beschleunigen, die Abstandsbestimmungen für alle Netzkpunkte durchführen (z. B. Summed Squared Distance und Hausdorff Distance), wurde folgende Vorgehensweise entwickelt: Anstatt alle Netzkpunkte zu bewerten, wird eine kleinere Anzahl an Netzkpunkten ausgewählt. Für diese wird der Abstand gemessen. Aufgrund dieser (wenigen) Messungen wird die Gesamtbewertung für alle Punkte abgeschätzt. Wird beispielsweise für die SSD-Berechnung 1% der Netzkpunkte verwendet, so wird das Ergebnis mit 100 multipliziert, um die Gesamtsumme abzuschätzen.

Es stellen sich folgende Fragen: Welche Netzkpunkte sollen verwendet werden? Sollten die Punkte fix ausgewählt werden oder sollen bei jeder Bewertung andere Punkte gewählt werden? Wie viele Punkte sind für eine verlässliche Abschätzung notwendig?

Versuche (siehe [38]) haben gezeigt, dass zufällig mit uniformer Verteilung ausgewählte Punkte verwendet werden können. Diese sollten beim Start der Optimierung fix ausgewählt werden, um eine hohe Korrelation zwischen Schätzwert und tatsächlicher Bewertung zu erhalten. Werden sie bei jeder Bewertung neu gewählt, ergibt sich keine Korrelation, da sich durch jede Punktauswahl ein nahezu konstanter Offset zur vollständigen Bewertung einstellt [38]. Besser als eine zufällige Punktauswahl ist die Verwendung jedes n -ten Punktes aus der Vertexliste. Da die Bauteile nach den in [123] beschriebenen Verfahren vernetzt wurden, kann davon ausgegangen werden, dass diese Punkte gleichmäßig über die Oberfläche verteilt sind. Experimente in [38] haben gezeigt, dass es bei der SSD-Berechnung ausreicht, 1% der Netzknoten zu verwenden, um eine Positionsbewertung durchzuführen. Die Rechenzeit ist annähernd proportional zur Anzahl der einbezogenen Punkte. Auf die Korrelation zwischen vollständigen und abgeschätzten beidseitigen Hausdorff-Abstand und hierzu durchgeführte Experimenten wird in 4.5.2 eingegangen.

4.4.3 Parallelisierung

Die Relative Positionierung lässt sich auf verschiedene Arten parallelisieren (siehe Bild 22): Verschiedene nichtideale Varianten ein und desselben Nominalbauteils lassen sich gleichzeitig und unabhängig voneinander positionieren. Bei einigen heuristischen Optimierungsverfahren können mehrere 6D-Positionen gleichzeitig ausgewertet werden, beispielsweise lassen sich die Positionen aller Partikel eines Partikelschwarms innerhalb einer Iteration parallel bewerten. Die aufsummierte Bewertungsfunktion lässt sich aufteilen in die einzelnen, voneinander unabhängigen Bewertungsfunktionen (z. B. Kollisionserkennung und SSD). Des Weiteren sind einige Bewertungsfunktionen selbst wiederum aufteilbar in viele unabhängige Einzelschritte (z. B. Abstandsmessungen für viele Punkte).

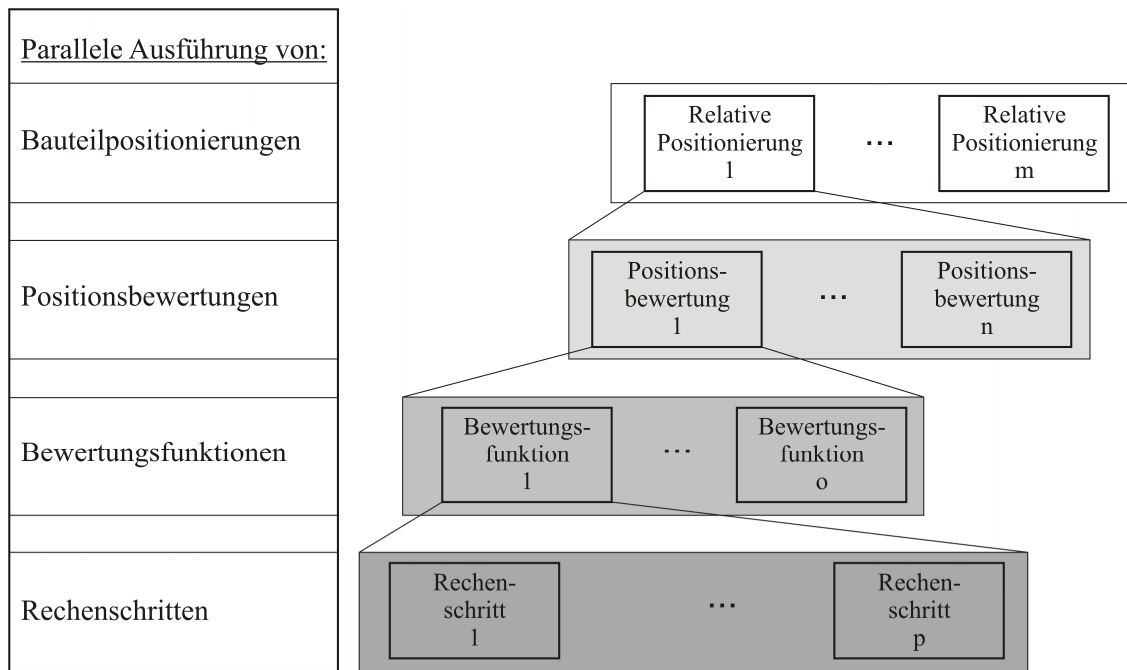


Bild 22: Parallelisierungsansätze für die Relative Positionierung

Implementiert wurde die Parallelisierung der Bauteilpositionierungen über parallel laufende Prozesse mit eigenen Speicherbereichen (Start über Batchskripte) sowie über parallele Positionsbewertungen durch MPI (siehe [38] und [39]). Parallele, unabhängige Prozesse ermöglichen eine einfache Parallelisierung der Positionierung für einen einzelnen Mehrkernrechner ohne Eingriff in die Programmlogik. MPI (siehe [44]) erlaubt die Parallelisierung sowohl auf einzelnen Rechnern als auch auf heterogenen Rechenclustern. Als MPI-Parallelisierungsart wurde das asynchrone Master-Slave-Model gewählt: Ein Steuerprozess (Master) verteilt Aufgabenpakete an die mitwirkenden Rechenkerne (Slaves). Die Slaves arbeiten die Pakete ab und liefern das Ergebnis an den Master, der wiederum neue Aufgaben verteilt.

4.5 Experimentelle Ergebnisse

Zur Evaluierung der entwickelten Positionierungsmethoden wurden einige Experimente mit verschiedenen Baugruppen durchgeführt. Sofern nicht anders angegeben beziehen sich die Laufzeiten auf einen Intel Core2 Duo 6600 mit 2.4 GHz und 3 GB RAM. Als Betriebssystem kam Windows XP (32 Bit), Service Pack 3 zum Einsatz.

4.5.1 Zero-Test Hausdorff-Abstand

Um zu demonstrieren, dass das Problem der relativen Positionierung grundsätzlich wie vorgeschlagen über heuristische Optimierungsverfahren lösbar ist, wurde das

entwickelte Positionierungsframework zunächst auf Positionierungsprobleme mit bekanntem globalem Optimum angewendet.

In [118] wurde ein nominaler Automobilquerträger (14.400 Dreiecke, 9.222 Vertices, siehe Bild 23) kopiert, verschoben und rotiert. Als Bewertungsfunktion wurde der beidseitige Hausdorff-Abstand verwendet, die ideale Verschiebung ist somit die inverse Transformation, der beste mögliche Abstand beträgt 0 mm, da die Form des Bauteils durch die Transformation nicht verändert wird. Die Optimierer Evolutionärer Algorithmus, Simulated Annealing und Partikelschwarmoptimierung durften jeweils 4000-mal die Bewertungsfunktion aufrufen. Die Auswertungen der Positionen nehmen, verglichen mit den Operationen des Optimierers, den Großteil der Rechenzeit in Anspruch. Die PSO erzielte die besten Durchschnittsergebnisse (Mittlerer Abstand der Endergebnisse 0,0000238 mm - Standardabweichung 0,0000431 mm), gefolgt von EA (0,0269 mm - 0,0328 mm) und SA (0,302 mm - 0,0441 mm). Grund für die guten Ergebnisse der PSO ist vermutlich das kontinuierliche „Durchstreifen“ des Suchraums. Für dieses einfache Problem ohne Umgebungsgeometrie lässt sich die Positionierung folglich zuverlässig durchführen, das globale Optimum wurde bei jedem Durchlauf nahezu exakt gefunden.

4.5.2 Korrelation zwischen vollständigem und abgeschätztem Hausdorffabstand

In [39] wurde bereits gezeigt, dass die Verwendung von 1% aller Vertices von *data* ausreicht, um die Summed Squared Distance mit ausreichender Genauigkeit abzuschätzen. Die Berechnung des beidseitigen Hausdorff-Abstands ist im Vergleich zu SSD noch aufwändiger, da hier zusätzlich die Abstände der Vertices von *model* zur Oberfläche von *data* berechnet werden. Zur Beschleunigung der Bewertungsfunktion wurde die Möglichkeit untersucht, nur einen geringen Prozentsatz der Vertices beider Netze in die Berechnung einzubeziehen. Da beim Hausdorff-Abstand nur der maximal gefundene Abstandswert zurückgeliefert wird, muss nach einer Abschätzung der gefundene Maximalwert nicht skaliert werden.

Als Testdatensatz wurde ein fein vernetzter nichtidealer Querträger (49.030 Dreiecke, 26.250 Vertices) auf einen PKW-Hauptboden (50.037 Dreiecke, 27.885 Vertices) positioniert, siehe Bild 23. Als *model* diente der ideale, aus dem CAD-Datensatz generierte Querträger mit 14.400 Dreiecken und 9.222 Vertices.

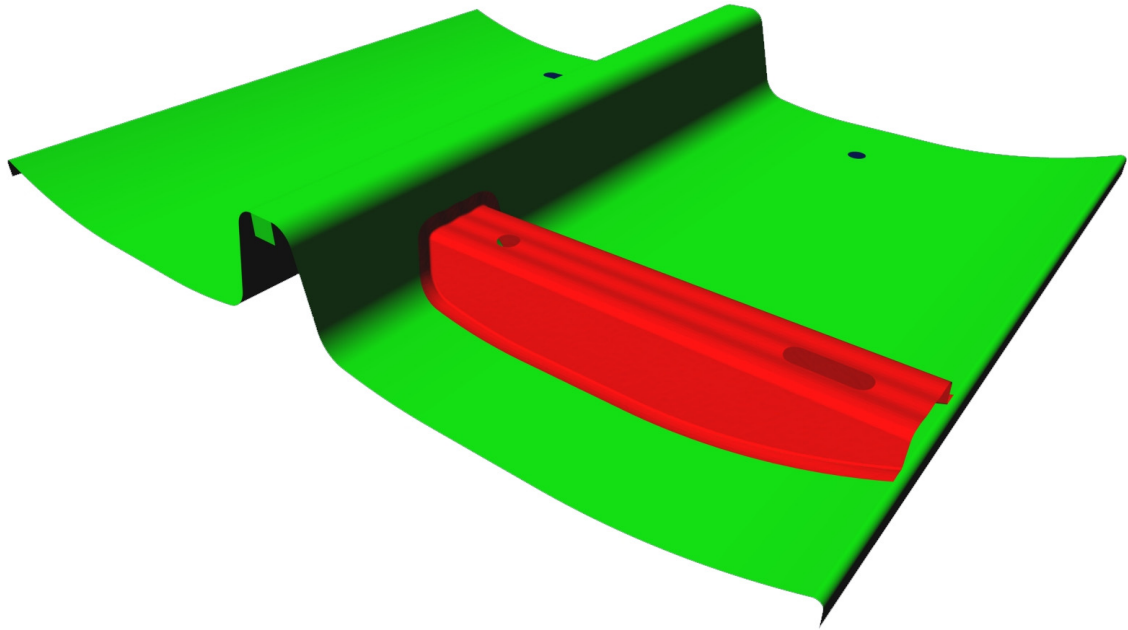


Bild 23: Positionierung eines nicht-idealen Querträgers (rot) auf einem PKW-Unterboden (grün)

Die Bewertungsfunktionen “Kollisionsvermeidung” und “beidseitiger Hausdorff-Abstand von p Prozent der Vertices” wurden für die Positionierung verwendet.

Es wurden drei verschiedene prozentuale Anteile (0,1%, 1% und 5%) der Eckpunkte zur Abschätzung gewählt. Für jeden Wert wurden zehn Optimierungsläufe mit jeweils 10.000 Funktionsauswertungen durch PSO durchgeführt. Für jeden abgeschätzten Abstandswert wurde zusätzlich der exakte Abstandswert (bei Verwendung von 100% der Vertices) berechnet und aus diesen Tupeln die Korrelation bestimmt (siehe Tabelle 3).

Tabelle 3: Auswertung der abgeschätzten Hausdorff-Bewertung

Prozentsatz p :	0,1%	1%	5%	100%
Verwendete Vertices (model/data)	9/26	92/263	460/1313	9.222/ 26.250
Korrelation	0,982	0,997	0,999	1
Endergebnis real (mm)	2,07 -2,42	1,65 -1,71	1,69 – 1,73	1,62 -1,63
Ø Rechenzeit (s)	47	78	178	1922

Die berechnete Korrelation ist zwar bereits bei der Verwendung von 0,1% der Vertices mit 0,982 relativ hoch. Um *data* dadurch ausreichend genau zu positionieren, reicht dies nicht aus. Die geschätzten Abstandswerte nach der Optimierung lagen für 10 Testläufe bei 1,44 mm, die tatsächlichen Abstandswerte dieser Ergebnisse lagen jedoch zwischen

2,07 mm bis 2,42 mm. Bei Verwendung aller Eckpunkte ließ sich der Querträger mit einem beidseitigen Hausdorff-Abstand von 1,62 mm platzieren, die abschätzende Positionierung ist somit bis zu 0,8 mm von einer sehr guten Positionierung entfernt.

Erst ab einer Verwendung von 1% der Vertices ist eine für die Toleranzanalyse dieser Baugruppe sinnvolle Positioniergenauigkeit im 0,1 mm – Bereich möglich. Aus den Experimenten ist erkennbar, dass die Positionierung über den Hausdorff-Abstand stark davon abhängt, welche Eckpunkte in den Abstandsmessungen betrachtet werden. Anders als bei der SSD-Berechnung wird nicht eine Gesamtsumme vieler Messungen zurückgeliefert, sondern nur der Abstandswert des am weitesten entfernten Punkts, daher reagiert die Bewertung sensibler auf das Verwenden bzw. Auslassen von bestimmten, kritischen Eckpunkten. Andererseits beschleunigt die Abschätzung des Hausdorff-Abstands die Positionierung sehr stark (z. B. Faktor 25 bei der Auswahl von 1% der Vertices). Die Geschwindigkeit ändert sich nicht linear aufgrund des Festplattenzugriffs beim Einlesen der Netze und Caching-Effekten.

Der Anwender muss daher entscheiden, welche Positionierungsungenauigkeit beziehungsweise welche Rechenzeit für den analysierten Anwendungsfall akzeptabel ist. Sind vereinzelt lokal auftretende, starke Punktabweichungen der simulierten Bauteilvarianten zu erwarten, muss ein hoher Anteil der Netzkpunkte ausgewertet werden.

4.5.3 Positionierung eines Kreuzziehteils auf einer ebenen Fläche

Als weiteres Experiment der Positionierung mit bekanntem Ergebnis wurde ein tiefgezogener Napf, das sog. Kreuzziehteil, aus CAD-Daten analysiert. Der nominale Napf aus 19.952 Dreiecken besitzt eine Höhe von $H_{\text{nominal}} = 53,7$ mm und wurde auf eine ebene Fläche (repräsentiert durch zwei Dreiecke) anhand der Bewertungsfunktionen “Kollisionsvermeidung“ und “Richtungsbewertung“ platziert.

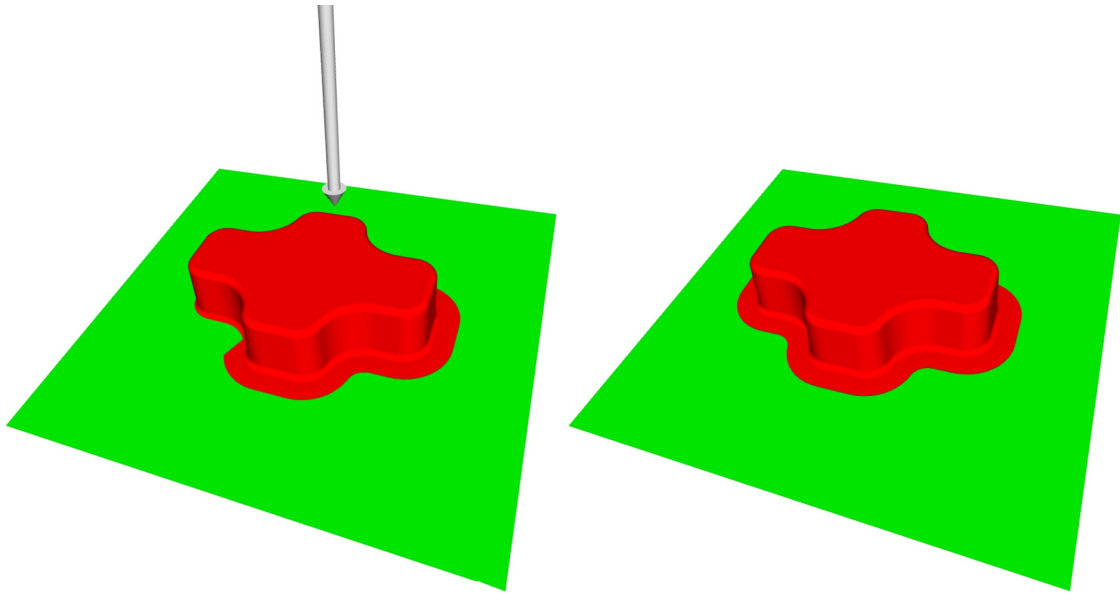


Bild 24: Positionierung eines verkippten Kreuzziehteils auf einer ebenen Fläche

Im durchgeführten Experiment wurde das Kreuzziehteil leicht verkippt, so dass es mit dem Boden (grün) kollidiert, siehe Bild 24, links. Zur Optimierung wurde die PSO angewandt, limitiert auf 10.000 Aufrufe der Bewertungsfunktionen; der Suchraum wurde auf ± 10 mm für die zulässigen Translationen und $\pm 1^\circ$ für zulässige Rotationen beschränkt. Anschließend wurde die Höhe H_{m1} des höchsten Eckpunkts jedes platzierten Bauteilnetzes vom Boden aus gemessen. Da bekannt ist, dass der ideale Napf eine Höhe von exakt 53,7 mm besitzt, kann so der durch die Positionierung entstandene Fehler gemessen werden. Der Mittelwert aller gemessenen H_{m1} wird als \bar{H}_{m1} , der höchste aufgetretene Wert von H_{m1} aller Durchläufe als $H_{m1}(\text{Max})$ bezeichnet. σ gibt die Standardabweichung des aufgeführten Messwertes an. Folgende Ergebnisse wurden bei der Durchführung von 100 Optimierungsläufen erzielt:

$$\bar{H}_{m1} = 53,7001608 \text{ mm}$$

$$\bar{H}_{m1} - H_{\text{nominal}} = 0,0001608 \text{ mm}, \sigma = 0,00004955 \text{ mm}$$

$$H_{m1}(\text{Max}) = 53,70035442 \text{ mm}$$

Mittlere Laufzeit = 7,9 Sekunden

Die ermittelten Werte, insbesondere der kleine durchschnittliche Positionierungsfehler, zeigen, dass die Relative Positionierung durch PSO für dieses Szenario bei sehr kurzer Rechenzeit zuverlässig Ergebnisse hoher Qualität liefert. Alle Bauteile wurden kollisionsfrei positioniert.

4.5.4 Zusammenbau zweier nominaler Kreuzziehteile

Als komplexere Positionierungsaufgabe wurde der virtuelle Zusammenbau von zwei nominalen Kreuzziehteilen mit Hilfe einer Vorrichtung betrachtet. Die Vorrichtung dient zur Ausrichtung der Bauteile nach dem 3-2-1-Prinzip (siehe Kapitel 4.3.4). Sie wurde aus dem CAD-System exportiert und besitzt 1.780 Dreiecke. Der Zusammenbau ist in zwei Schritte unterteilt. Zuerst wird ein ideales Kreuzziehteil auf der Vorrichtung positioniert (Bild 25, links). Als Bewertungsfunktionen dienen “Kollisionsvermeidung“ und “Richtungsbewertung“ mit dem Vektor $\vec{v} = (-2, -3, 1)$, der als weißer Pfeil in Bild 25 dargestellt ist. Nach diesem Schritt wird das um 180° rotierte Kreuzziehteil (blau) unter Verwendung der gleichen Bewertungsfunktionen auf dem Unterteil (rot) platziert (Bild 25, rechts).

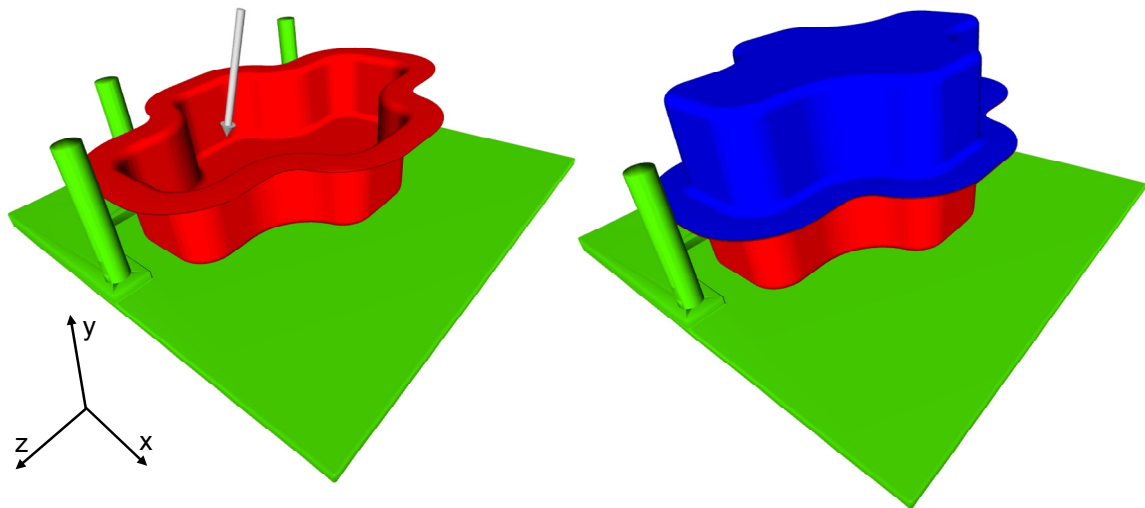


Bild 25: Relative Positionierung einer Baugruppe aus zwei Kreuzziehteilen

Das zweistufige Experiment wurde 100-mal durchgeführt, danach wurde jeweils der Abstand des höchsten Vertex des oberen Napfes zum Boden (bezeichnet als H_{m2}) gemessen.

$$\bar{H}_{m2} = 107,412236 \text{ mm}$$

$$\bar{H}_{m2} - H_{ideal2} = 0,012236 \text{ mm}, \sigma = 0,00683 \text{ mm}$$

$$H_{m2}(\text{Max}) = 107,447043 \text{ mm}$$

Mittlere Laufzeit (Napf 1 auf Vorrichtung): 12,4 Sekunden

Mittlere Laufzeit (Napf 2 auf Napf 1): 74,8 Sekunden

Die relativ lange Laufzeit der Positionierung von Napf 2 auf Napf 1 ergibt sich daraus, dass im Flanschbereich eine hohe Anzahl von fast parallelen Dreiecken

aneinandergeschoben werden müssen. Die Kollisionserkennung kann diese Dreiecke somit nicht frühzeitig über eine Trennebene aus der Berechnung ausschließen.

Die erzielte Ergebnisqualität ist für eine Baugruppe dieser Größenordnung mit einem durchschnittlichen Fehler von 0,012 mm (entspricht 0,011% der Gesamthöhe) und einem maximal aufgetretenen Fehler von 0,045 mm sehr gut für Toleranzanalysen verwendbar. Alle Bauteile wurden durch die Optimierung kollisionsfrei positioniert.

4.5.5 Relative Positionierung simulierter Kreuzzieh-Bauteile

Um realistischere Ergebnisse zu erhalten als bei rein zufälliger Geometrieveränderung innerhalb der Toleranzgrenzen, lässt sich das Fertigungsverfahren der Bauteile in die Analyse mit einbeziehen. In der Veröffentlichung [115] wurde daher die Fertigung und der virtuelle Zusammenbau der bereits beschriebenen Kreuzziehteile simuliert. Innerhalb der Finite-Elemente-Simulation (FE-Simulation, siehe Bild 26) des Fertigungsprozesses “Tiefziehen“ wurden die Parameter Blechdicke, Blechposition, Niederhalterkraft, Reibungskoeffizient zwischen Blech und gesamtem Werkzeug sowie Materialeigenschaften des Stahlblechs variiert. Bild 27 zeigt eine Variante des Kreuzziehteils, die unter Minimierung des Summed Squared Distance mit dem Nominalteil überlagert dargestellt wird. Sichtbar sind die geometrischen Abweichungen z. B. durch Rückfederung und Blechausdünnung.

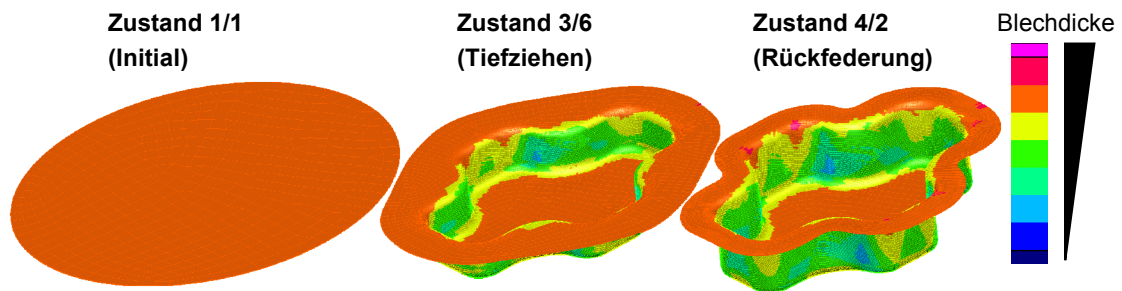


Bild 26: Zwischenschritte der FE-Simulation des Tiefziehprozesses (siehe [115])

Ziel des Experiments war es, den Zusammenbau von jeweils zwei realistischen Tiefziehteilen zu simulieren. Eine Monte-Carlo-Simulation aller Fertigungs- und Materialparameter und eine Verwendung von zwei zufällig erzeugten Tiefziehteilen für jede Relative Positionierung war aus zeitlichen Gründen nicht möglich. Ein einzelner FE-Simulationslauf zur Erzeugung einer Bauteilvariante benötigt etwa 1,5 h. Um trotz weniger Samples (Bauteilvarianten) statistische Aussagen treffen zu können, wurde Latin Hypercube Sampling (siehe [84]) verwendet. Auf diese Weise wurden 74 Varianten des Kreuzziehteils erstellt, die jeweils aus circa 67.000 Dreiecken bestehen.

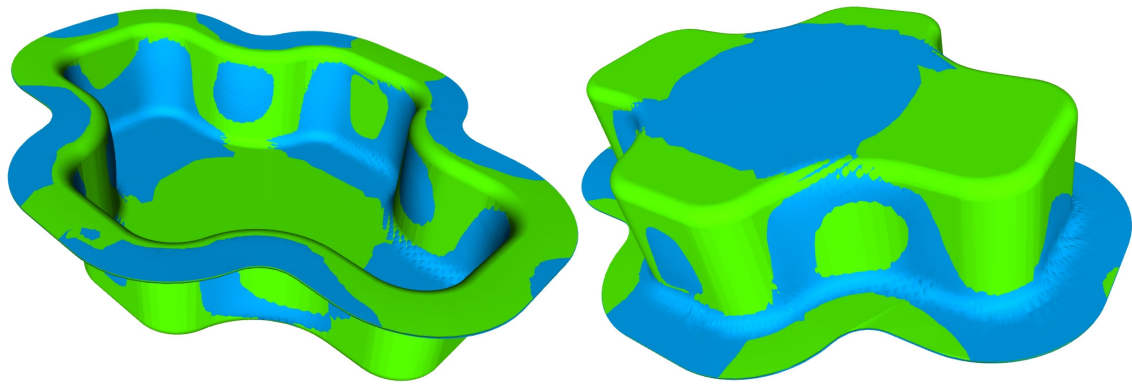


Bild 27: Vergleich eines FE-simulierten Kreuzziehteils (blau) mit der Idealgeometrie (grün): Beide Bauteiloberflächen sind überlagert dargestellt

Zunächst wurden alle 74 Näpfe auf der Vorrichtung positioniert. Auf jeden positionierten unteren Napf wurden wiederum alle 74 oberen Näpfe positioniert, so dass $74 \cdot 74 = 5476$ Baugruppen entstanden.

Ein statistisch zuverlässigeres Ergebnis würde sich ergeben, wenn für jede simulierte Baugruppenvariante sowohl Unter- als auch Oberteil zufällig erzeugt werden. Auf diese Weise wären beide Teile voneinander unabhängig. Da eine große Anzahl an Baugruppen (>1000) zur statistischen Analyse notwendig ist, konnte dies aus Rechenzeitgründen nicht durchgeführt werden. Bei der verwendeten Vorgehensweise wird in Kauf genommen, dass ein einzelnes, extremes FE-Simulationsergebnis großen Einfluss auf das Endergebnis hat, da es mehrfach in der Analyse verwendet wird.

Die angewandten Bewertungsfunktionen (Kollisionsvermeidung und Richtungsbewertung) der Bauteilpositionierung sind identisch mit den bereits beschriebenen Einstellungen zur Positionierung der idealen Baugruppe (siehe Kapitel 4.5.4). Die Partikelschwarmoptimierung führte jeweils 50.000 Aufrufe der Bewertungsfunktion pro Kreuzziehteil aus.

Gemessen wurde die Gesamthöhe der Baugruppe in der Mitte des oberen Napfes, bezeichnet als H_{m2FE} . Da die Höhe als Zweipunkmaß definiert ist, ist diese Messweise zulässig, siehe [50]. Folgende Werte wurden ermittelt:

$$\bar{H}_{m2FE} = 108,84 \text{ mm}, \sigma = 0,08 \text{ mm}$$

$$H_{m2FE} (\text{Max}) = 109,15 \text{ mm}$$

Mittlere Laufzeit (Napf 1 auf Vorrichtung): 38,2 Sekunden

Mittlere Laufzeit (Napf 2 auf Napf 1): 58,5 Sekunden

Bei 10 der 5476 Positioniervorgänge (0,18%) der oberen Kreuzziehteile fand der Optimierer keine kollisionsfreie Position, obwohl diese existiert. Diese

Optimierungsläufe wurden abermals gestartet und lieferten im zweiten Durchlauf kollisionsfreie Lösungen. Die kollisionsfreien Lösungen wurden zur Ermittlung des Mittelwertes verwendet.

Die Versuche in [115] haben ergeben, dass sich die Ergebnisqualität nach 20.000 Aufrufen der Bewertungsfunktion nur noch unwesentlich verbessert. Daher wurde die Zahl der Bewertungsaufrufe auf 20.000 reduziert. Um das Volumen des Parameterraums zu verringern, in dem der obere Napf mit unterem Napf bzw. Vorrichtung kollidiert, wurde der Suchraum leicht verschoben. Das Experiment wurde mit den angepassten Einstellungen erneut durchgeführt. Es ergaben sich folgende Ergebnisse:

$$\bar{H}_{m2FE} = 108,83 \text{ mm}, \sigma = 0,08 \text{ mm}$$

$$H_{m2FE} (\text{Max}) = 109,08 \text{ mm}$$

Mittlere Laufzeit (Napf 1 auf Vorrichtung): 25,6 Sekunden

Mittlere Laufzeit (Napf 2 auf Napf 1): 45,0 Sekunden

Die Ergebnisse sind nahezu identisch mit denen des vorhergegangenen, aufwändigeren Experiments, die gemessenen Höhenwerte unterscheiden sich im direkten Vergleich durchschnittlich um 0,0036 mm. In nur 0,3% der Durchläufe unterscheidet sich die Höhe um mehr als 0,05 mm. Daraus lässt sich für diesen Versuchsaufbau das Fazit ableiten, dass die Optimierung für eine gegebene Problemstellung zuverlässig ähnlich gute Ergebnisse berechnet und nicht bei jedem Lauf zufällig völlig unterschiedliche Werte liefert.

Bei nahezu gleichen Endergebnissen wurden durch die angepassten Parametereinstellungen alle Näpfe korrekt kollisionsfrei positioniert. Die Laufzeit wurde durch die niedrigere Anzahl an Bewertungsaufrufen auf 67% der ursprünglichen Rechenzeit reduziert.

Neben der geometrischen Analyse der Baugruppe wurde eine konventionelle CAT-Analyse durchgeführt (siehe [115]). Der Versuchsaufbau wurde in VisVSA abgebildet. Dabei wurde darauf geachtet, den Versuchsaufbau des FE-Experiments (Positionierungsschema, Messungsdefinition etc.) so genau wie möglich zu übertragen. Die Verteilungskurven der abweichungsbehafteten Parameter wurden bei der kommerziellen Toleranzsimulation durch die Toleranzdefinition entsprechend der Empfehlungen der Fachliteratur festgelegt, da keine Informationen über die realen

Wertevertellungen vorliegen. Die Bauteilgeometrie wurde ins CAT-System importiert, Features abgeleitet, Toleranzen definiert und die Zusammenbau-Operationen festgelegt. Vergleicht man das Qualitätskriterium „Gesamthöhe der Baugruppe“, unterscheiden sich die Mittelwerte der beiden Simulationstypen CAT und FE mit Relativer Positionierung um beachtliche 1,35 mm.

Die Spezifikationsgrenzen der Baugruppenhöhe (siehe Bild 28) entstammen den funktionalen Anforderungen an die Baugruppe und der darauf basierenden Tolerierung. Die CAT-Simulation berechnet einen Mittelwert der Baugruppe von 107,49 mm (Standardabweichung 0,22 mm), wonach alle erzeugten Baugruppen innerhalb der Spezifikationsgrenzen liegen. Die Simulation der Fertigung und Bauteilmontage ergibt hingegen einen Mittelwert von 108,84 mm (Standardabweichung 0,08 mm), d.h. nahezu alle virtuell montierten Baugruppen liegen über dem oberen Spezifikationslimit.

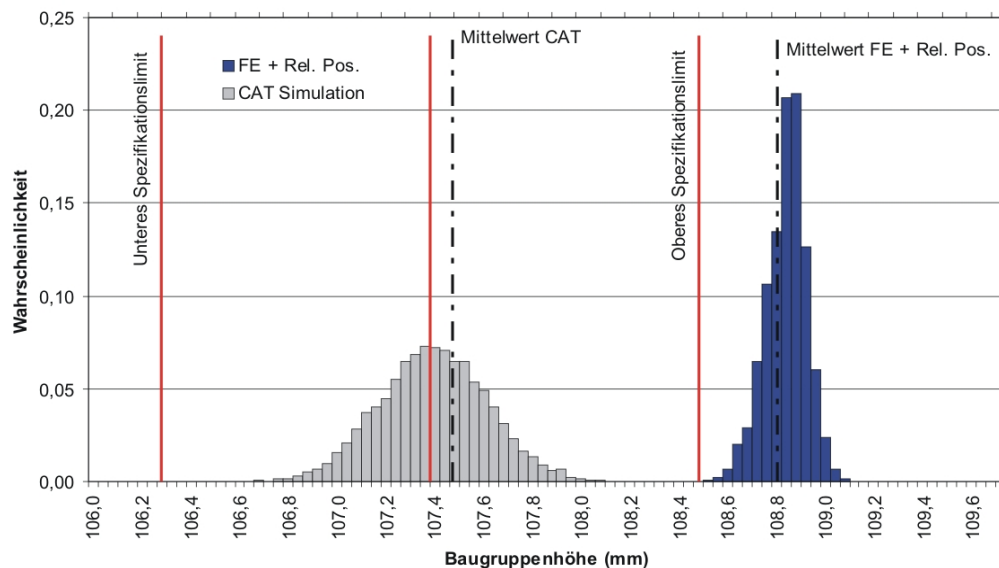


Bild 28: Vergleich der Baugruppenhöhen bei CAT-Simulation und Relativer Positionierung [115]

Um zu überprüfen, welche Simulationsalternative näher an der Realität liegt, wurden Tiefziehteile gefertigt und optisch vermessen. Die Messungen zeigten eine gute Übereinstimmung mit den durch FE-Simulation erzeugten Bauteilvarianten und entsprachen den FE-Netzen sowohl in Spannbreite der Abweichungen als auch bei der auftretenden Mittelwertverschiebungen. Bei der CAT-Simulation wurden die Schwankungsbreiten (beispielsweise der Ebenheitsabweichungen an der Bauteilunterseite und im Flanschbereich) zu klein angenommen. Details zum Abgleich von Simulation und Versuch wurden in [113] veröffentlicht.

Durch die aufeinander aufbauende Verwendung von Fertigungssimulation und Relativer Positionierung konnten somit noch vor der realen Fertigung Aussagen über zu

erwartende Produkteigenschaften getroffen werden, die sich bei der Überprüfung als richtig erwiesen haben. Die vorgeschlagene Vorgehensweise bietet daher einen deutlichen Mehrwert gegenüber der verbreiteten CAT-Simulation.

4.6 Fazit und Ausblick Relative Positionierung

Es wurde gezeigt, dass das Problem der Relativen Positionierung nicht-nominaler Bauteile mit dem gewählten Ansatz der heuristischen Optimierung verlässlich lösbar ist: Die berechneten Positionen für Tests mit bekannter Idealtransformation lagen sehr nahe am globalen Optimum und lieferten zuverlässig gute Ergebnisse. Die Positionierungsergebnisse für simulierte, nichtideale Bauteile wurden visuell überprüft und waren allesamt plausibel. Die Bauteile lagen auf der umgebenden Baugruppe auf. Es gab keine Durchdringungen oder Schweben von Bauteilen. Eine offensichtlich bessere Position hinsichtlich der Bewertungsfunktionen war nicht auszumachen.

Durch den modularen Aufbau des Optimierungsframeworks lassen sich flexibel weitere Bewertungsfunktionen ergänzen, um so beispielsweise ein weiteres spezielles Positionierungsschema nachzubilden.

Die Ergebnisse der Relativen Positionierung sind visualisierbar und können zusätzlich beliebig vermessen werden, da die gesamte Baugruppengeometrie vorliegt. Im Gegensatz zu den verbreiteten CAT-Techniken müssen Problemstellen der betrachteten Baugruppe vor der Analyse noch nicht bekannt sein. In Kapitel 6 werden anwendbare Visualisierungs- und Messmethoden sowie Verfahren zur haptischen Analyse der Geometriedaten im Detail vorgestellt.

Durch Einbeziehung von Fertigungssimulationen lassen sich Abweichungseffekte in ihrer zu erwartenden Form und Größe genauer vorhersagen, als dies bisher durch Computer Aided Tolerancing Systeme möglich ist. Gleichzeitig sinkt der Aufwand, den der Benutzer in die Zusammenstellung der Simulationsdaten investieren muss.

Erkauft werden die genannten Vorteile durch einen erhöhten Rechenaufwand. Durch die genannten Beschleunigungstechniken [119], [39] und angepasste Samplingverfahren (siehe [115]) eignet sich die Relative Positionierung bereits zur Analyse praktischer Problemstellungen.

Noch nicht betrachtet wurde bisher die Montage statisch überbestimmter Bauteile. Der Benutzer muss stets eine Montagereihenfolge angeben. Dies könnte dahingehend erweitert werden, dass automatisch der Montagereihenfolge mit den geringsten Abweichungen relevanter Maße bestimmt wird. Des Weiteren könnte die Relative

Positionierung dahingehend erweitert werden, dass mehrere Bauteile gleichzeitig als beweglich betrachtet werden. Dabei ist das Problem zu lösen, dass die Dimensionalität des Suchraums weiter ansteigt.

5 Pfadplanung

Im vorhergehenden Kapitel wurde beschrieben, wie sich über heuristische Optimierungsverfahren die Endposition abweichungsbehafteter Bauteile bestimmen lässt. An dieser Position ist das betrachtete Bauteil (falls geometrisch möglich) frei von Überschneidungen mit der umgebenden Baugruppe. Wurde eine Endposition mit dieser Eigenschaft gefunden, ist eine weitere Analyse notwendig: Zusätzlich muss überprüft werden, ob der gesamte Montagevorgang kollisionsfrei durchführbar ist (siehe Bild 29).

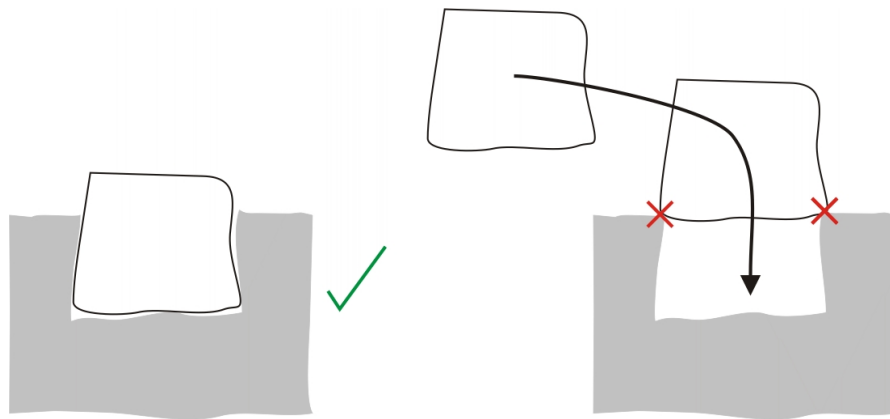


Bild 29: Eine kollisionsfreie Endposition ist vorhanden (links), der Montagevorgang ist jedoch nicht durchführbar (rechts)

Der einfachste Ansatz zur Überprüfung der Montierbarkeit besteht darin, einen einzigen, fixen Montagepfad vorzugeben und für viele nicht-nominale Bauteil- und Umgebungsvarianten Kollisionsprüfungen auf diesem Pfad durchzuführen.

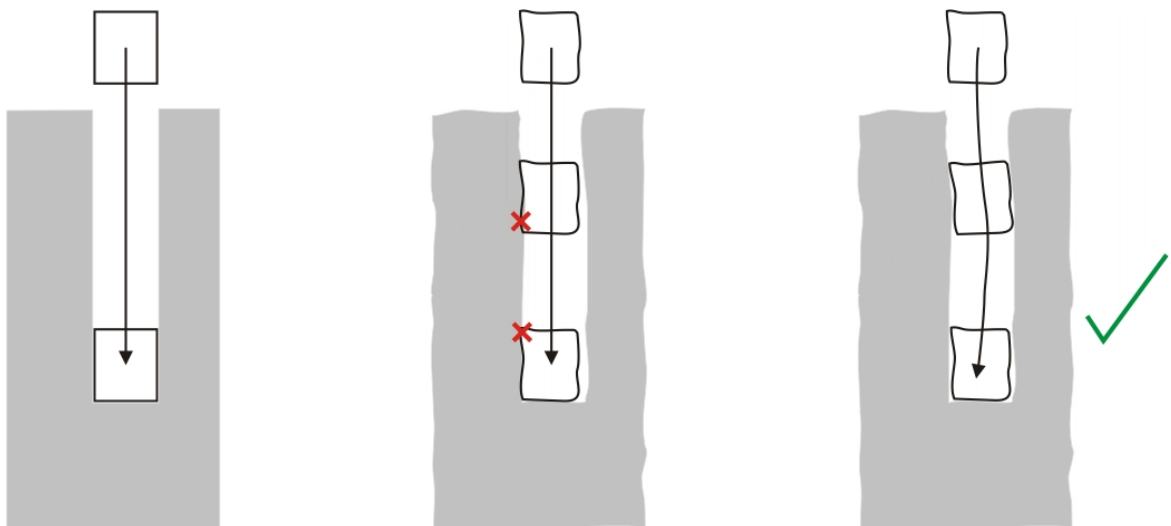


Bild 30: Kollisionsanalyse mit fixem (Mitte) und neu berechnetem (rechts) Montagepfad

Diese Vorgehensweise bezeichnet jedoch Bauteile als „nicht montierbar“, die durch einen alternativen Montagepfad kollisionsfrei montierbar wären, siehe Bild 30. Daher

ist es nötig, eine Pfadsuche für jedes analysierte nichtideale Bauteil durchzuführen. Im folgenden Unterkapitel werden die Grundlagen der automatischen Pfadplanung und die wichtigsten bestehenden Forschungsansätzen auf diesem Gebiet zusammengefasst.

5.1 Betrachtung bestehender Verfahren

Im folgenden Text werden die grundlegenden Begriffe, Konzepte und Algorithmen auf dem Gebiet der Pfadplanung erläutert. Diese stellen die Basis für die Entwicklung einer Pfadplanung zur Toleranzanalyse dar.

5.1.1 Grundbegriffe der Pfadplanung

Unter dem Begriff *Pfadplanung* (*path planning*) versteht man das rein geometrische Problem, für ein bewegliches Objekt (oft als *robot* bezeichnet) einen kollisionsfreien Pfad zu berechnen, der feste Hindernisse umgeht. Vorgegeben ist meist eine Start- und Zielposition bzw. ein Zielgebiet. Der Begriff *Bewegungsplanung* (*motion planning*) hingegen wird bei Problemen verwendet, die zusätzlich Zeit, dynamische Randbedingungen, Sensordatenverarbeitung etc. beinhalten (siehe [69]). Bei der Toleranzanalyse spielen solche zusätzlichen Randbedingungen keine Rolle, daher wird nur die geometrische Montierbarkeit untersucht. Die Aufgabe ist also ein reines Pfadplanungsproblem.

Ein *vollständiger Pfadplaner* (*complete path planner*, siehe [69], [73]) berechnet einen kollisionsfreien Pfad, falls dieser existiert und kann auch zeigen, dass kein Pfad existiert. Für das in der Toleranzanalyse untersuchte Problem ist kein universeller Ansatz bekannt, der in der Lage ist zu beweisen, dass kein Pfad existiert. Die veröffentlichten Ansätze zum Beweis der Nicht-Montierbarkeit sind meist auf bestimmte Geometrietypen beschränkt (siehe [7]) oder auf wenige Freiheitsgrade beschränkt (siehe [141]).

Ein *probabilistisch vollständiger Pfadplaner* (*probabilistically complete path planner*, siehe [73]) findet einen kollisionsfreien Pfad in endlicher Zeit, falls dieser existiert. Falls kein Pfad existiert, läuft der Algorithmus theoretisch unbegrenzte Zeit weiter. Im praktischen Einsatz bedeutet dies, dass ein probabilistisch vollständiger Planer bis zu einem vorgegebenen Abbruchkriterium, z. B. eine maximale Zeitspanne, Berechnungen durchführt. In dieser Zeit wird entweder eine Lösung gefunden, oder man geht davon aus, dass kein kollisionsfreier Pfad existiert. Bewiesen wird dies jedoch nicht, da die Berechnung abgebrochen wurde.

Der Zustand des bewegten Objekts mit n Freiheitsgraden (*degrees of freedoms*, DoFs) wird bei der Pfadplanung als Punkt im n -dimensionalen Konfigurationsraum (*configuration space*) beschrieben. Eine Konfiguration eines Bauteils bei der Montage in 3D entspricht einem Punkt in 6D, der die Bauteilverschiebung und die Bauteilrotation beschreibt. Der gesamte Konfigurationsraum lässt sich unterteilen in den Raum der erreichbaren Punkte (*free space*, C_{free}) und den Raum der durch Hindernisse blockierten Punkte (*obstacle space*, C_{obs}), siehe [69], [73]. Bild 31 zeigt symbolisch einen unterteilten Konfigurationsraum mit punktförmiger Start- und Zielkonfiguration.

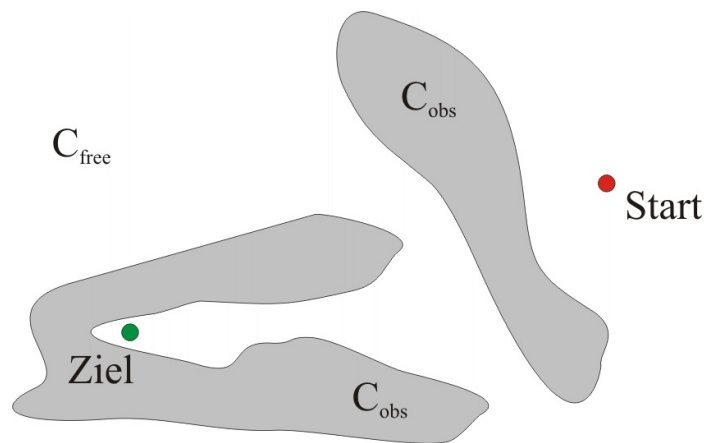


Bild 31: Unterteilung des Konfigurationsraums in C_{free} und C_{obs}

Als *lokalen Planer* bezeichnet man ein Verfahren, das zwei Punkte im Konfigurationsraum direkt miteinander verbindet. Lokale Planer sollten deterministisch und sehr schnell sein ([59]), damit der erzeugte Pfad nicht gespeichert werden muss, da er problemlos neu generierbar ist. Beispiele für lokale Planer sind lineare Parameterinterpolation ([59]) oder sphärische lineare Interpolation (siehe Kapitel 1.4.4). Der durch lokale Planer erzeugte Pfad wird an diskreten Zwischenpositionen auf Kollisionen überprüft, so dass eine maximale Schrittweite δ eingehalten wird.

Steuerparameter sind diejenigen Variablen, die bei der Steuerung eines Systems vom Benutzer frei gewählt werden können, bei einem Automobil beispielsweise Lenkrichtung und Beschleunigung. Systeme, bei denen die Anzahl der Freiheitsgrade gleich der Anzahl der Steuerparameter ist, heißen *holonom*. Die in den folgenden Ansätzen betrachteten Systeme werden als holonom angenommen. Jeder beliebige Pfad ist prinzipiell für den *robot* zulässig. *Nicht-holonome Systeme* besitzen eine geringere Anzahl an Steuerparameter als Freiheitsgrade. Ein Automobil ist beispielsweise ein nicht-holonomes System, da es drei Freiheitsgrade besitzt (x-, y-Koordinate, Orientierung), aufgrund der beiden Steuerparameter (Lenkrichtung, Beschleunigung)

aber keine beliebigen Bewegungen (z. B. paralleles Einparken) durchführen kann. Sind die erreichbaren Beschleunigungen des bewegten Objekts beschränkt, ist das System ebenfalls nicht-holonom.

5.1.2 Bestehende Ansätze

In den 1980ern waren die beiden Pfadplanungsansätze „Approximative Zellzerlegung“ und „Potentialfeldmethode“ dominant. Beide Ansätze sind jedoch nicht in der Lage, dreidimensionale Montageaufgaben zu lösen. Sie lassen sich nur für Aufgaben mit weniger als sechs Freiheitsgraden einsetzen, siehe [59] und ([69].

Erst in den 1990ern wurden randomisierte Planungsalgorithmen entwickelt, die in der Lage sind, höherdimensionale Probleme zu lösen. Zwei dieser Ansätze (Probabilistic Roadmap und Rapidly Exploring Random Tree) werden im Folgenden detailliert vorgestellt, da sie als Grundlage für die selbstentwickelte Montagesimulation tolerierter Baugruppen dienen.

Der von L. Kavraki et al. vorgestellte randomisierte **Probabilistic Roadmap Algorithmus** (PRM, siehe [59]) besteht aus zwei Phasen:

Lernphase

In der Lernphase werden zufällige Konfigurationen erzeugt und durch den lokalen Planer verbunden, um die Konnektivität des Konfigurationsraumes näherungsweise abzubilden. Der Algorithmus startet mit einem leeren, ungerichteten Graphen R , bestehend aus einer Menge an Knoten N (Konfigurationen) und einer Menge an Kanten E (Verbindungen, Pfade). Zufällig wird eine neue Konfiguration c erzeugt und zu N hinzugefügt. Nachfolgend wird basierend auf einer Abstandsmetrik eine Menge von Knoten N_c aus der Nachbarschaft von c gewählt. Anhand eines einfachen, deterministischen lokalen Planers wird versucht, c mit allen Knoten aus N_c zu verbinden. Falls der lokale Planer die Konfigurationen kollisionsfrei verbinden kann, wird eine neue Kante zwischen dem Nachbarn und c gespeichert. Dies geschieht jedoch nur, falls beide sich nicht in der gleichen Zusammenhangskomponente befinden, siehe Bild 32.

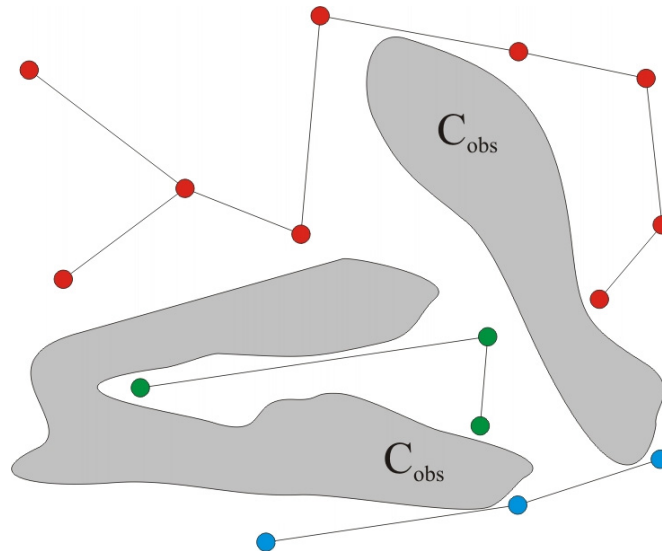


Bild 32: Konfigurationsraum mit drei Zusammenhangskomponenten (rot, grün und blau)

Danach wird versucht, den Graphen durch das Generieren zusätzlicher Knoten in schwierigen Bereichen zu verbessern. Zur Optimierung der Konnektivität existieren zahlreiche Ansätze. Die Lernphase wird so lange ausgeführt, bis die Konnektivität des Konfigurationsraumes durch die verbundenen Konfigurationen ausreichend gut abgebildet ist. Der aufgebaute Graph wird als *Roadmap* bezeichnet. Im gezeigten Beispiel (Bild 33) besteht die *Roadmap* nach Abschluss der Lernphase aus einer einzigen Zusammenhangskomponente, dies muss aber nicht zwingend der Fall sein.

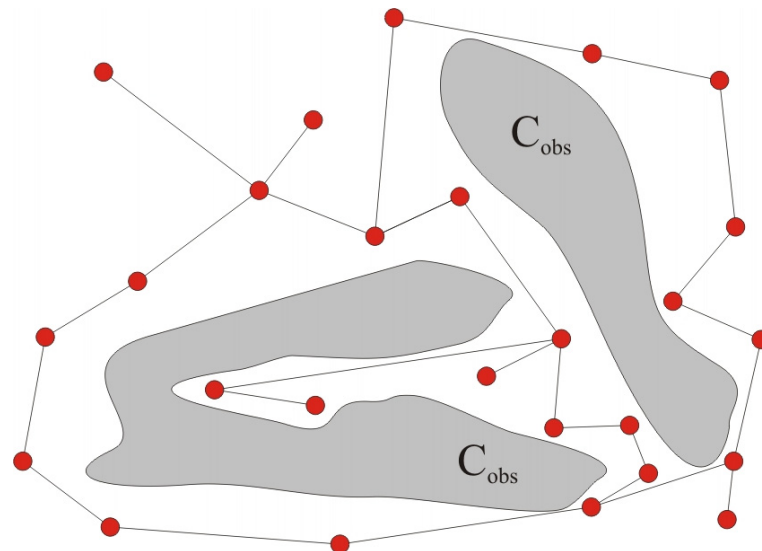


Bild 33: Der Konfigurationsraum ist nach der Lernphase durch eine zusammenhängende Roadmap abgebildet

Abfragephase

In der Abfragephase werden die (vom Benutzer nachträglich angegebene) Startkonfiguration und Zielkonfiguration mit der in der Lernphase aufgebauten Roadmap verbunden, vgl. gestrichelte Linien in Bild 34. Falls es möglich ist, beide Konfigurationen mit der gleichen Zusammenhangskomponente zu verbinden, existiert ein Pfad zwischen Start und Ziel. Der kürzeste Pfad kann durch Graphensuche, beispielsweise durch den A*-Algorithmus, wie in ([46]) beschrieben, berechnet werden, siehe fett markierte Kanten in Bild 34.

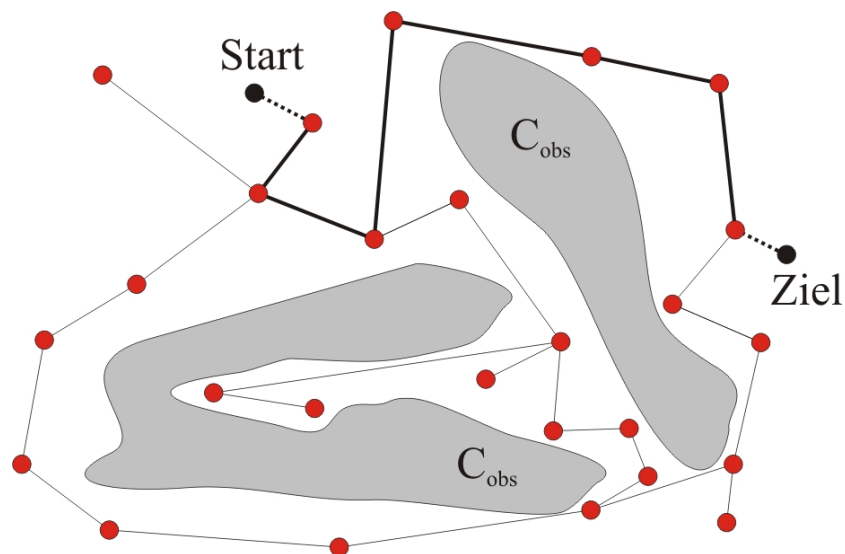


Bild 34: Pfadsuche von der Start- zur Zielkonfiguration

Verschiedene Techniken wurden vorgeschlagen, um das Sampling von langen, engen Passagen im Suchraum zu verbessern, da diese Durchgänge besonders schwierig zu finden sind:

In [4] wird der *robot* auf das Zentrum von umgebenden Hindernissen gelegt. In unterschiedliche Richtungen wird der *robot* vom Hindernis wegbewegt, es wird binär nach einem Punkt auf der Oberfläche von C_{obs} gesucht, an dem keine Kollisionen mehr auftreten. Die so gefundenen Punkte im Konfigurationsraum werden verwendet, um dessen Konnektivität besser abzubilden. Das Verfahren wird als *obstacle-based probabilistic roadmap method* (OBPRM) bezeichnet.

Durch das Zulassen einer geringen Eindringtiefe des *robots* in die Umgebungsgeometrie wird das Pfadplanungsproblem in [49] zunächst vereinfacht. Kollidierende Knoten im Originalproblem werden anschließend durch Abtasten der näheren Umgebung nach C_{free} geschoben.

Die Veröffentlichung [11] schlägt einen hybriden Ansatz vor, bei dem Benutzereingaben über ein Haptik-Gerät als Hilfe für den Pfadplanungsalgorithmus verwendet werden. Durch die groben, ggf. kollidierenden manuellen Pfadeingaben ist das Verfahren schneller in der Lage, schwierige Pfadplanungsaufgaben zu lösen. Eines der vorgeschlagenen Verfahren besteht darin, einen durch den Haptik-Operator erzeugten, kollidierenden Anfangspfad zu nutzen, um so lange iterativ vereinfachte Varianten des Problems zu lösen, bis schließlich das ursprüngliche Problem gelöst werden kann.

Um durch das Speichern von wenigen Konfigurationen bereits eine hohe Überdeckung von C_{free} zu erreichen, schlägt [89] einen sichtbarkeits-basierten PRM-Algorithmus (PRM-Vis) vor. Zufällige, kollisionsfreie Konfigurationen werden nur abgespeichert, wenn sie nicht direkt über einen lokalen Planer mit anderen gespeicherten Knoten verbunden werden können (*Guard-Knoten*) oder wenn sie zwei nicht verbundene Komponenten verbinden (*Connection-Knoten*). Durch das Verfahren reduzieren sich Speicherbedarf und Rechenzeit im Vergleich zum Basis-PRM beträchtlich.

Die bisher beschriebenen PRM-Techniken eignen sich wegen der einmal in der Lernphase aufgebauten *Roadmap* dazu, viele Suchanfragen für den betrachteten Konfigurationsraum sehr schnell zu beantworten. Das PRM-Verfahren wird daher als *Multiple Query*-Verfahren bezeichnet.

Im Gegensatz dazu stellt der **Rapidly Exploring Random Tree** Algorithmus (RRT, siehe [70]) ein erfolgreiches, weit verbreitetes *Single Query*-Verfahren dar. Der Basis-RRT-Algorithmus arbeitet wie folgt:

Ausgehend von einer Startkonfiguration baut der Algorithmus einen Baum auf, indem zufällige, neue Konfigurationen x_{rand} erzeugt werden. Zu x_{rand} wird die Konfiguration $x_{\text{near}} \in N$ gesucht, die entsprechend der verwendeten Metrik am nächsten an x_{rand} liegt. Von x_{near} aus wird ein kleiner Schritt in Richtung x_{rand} durchgeführt, so dass eine neue, kollisionsfreie Konfiguration x_{new} entsteht. Diese wird mit dem bestehenden Baum verbunden, siehe Bild 35. Die soeben beschriebene Erweiterung des Baums wird als *Extend-Operation* bezeichnet.

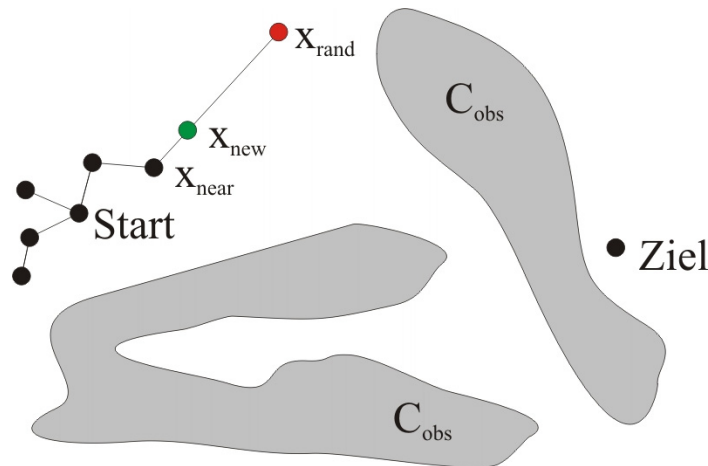


Bild 35: Erweiterung des RRT-Baumes in Richtung x_{rand} (rot) um eine Konfiguration (grün)

Anschließend wird überprüft, ob die Zielkonfiguration von x_{new} ausgehend direkt (durch einen lokalen Planer) erreichbar ist. Der Baum wird so lange erweitert, bis die Zielkonfiguration erreicht wurde oder ein Abbruchkriterium (z. B. maximale Laufzeit) erfüllt wurde. Von diesem Basis-RRT wurden zahlreiche Varianten abgeleitet. In [71] und [68] wurde das Verfahren RRT-Connect vorgestellt. Hier wird von x_{near} aus durch den lokalen Planer eine Transformation so weit in Richtung x_{rand} durchgeführt, bis eine Kollision auftritt, oder x_{rand} erreicht ist (Connect-Operation). Das Verfahren liefert nach [71] für holonome Pfadplanungsprobleme bessere Ergebnisse als der Basis-Algorithmus. Bidirektionale Verfahren generieren Bäume sowohl von der Start- als auch der Zielkonfiguration aus und versuchen, die Äste beider Bäume zu verbinden, siehe beispielsweise [72] und [71]. Als erfolgreichstes Verfahren für holonome Probleme wird in [71] RTT-ExtCon genannt, das die jeweiligen Bäume mit der Extend-Operation erweitert und über die Connect-Operation versucht, die Bäume zu verbinden. Die Veröffentlichung [32] beschreibt die Verwendung eines einseitigen RRT-Verfahrens zur Demontagesimulation von Bauteilen. Einige Modifikationen wurden am Algorithmus vorgenommen, um die Erfolgswahrscheinlichkeit zu erhöhen: Der lokale Planer wendet anfangs eine sehr große Schrittweite δ bei der Interpolation zwischen Konfigurationen an. Enge Passagen können dadurch ggf. zunächst übersprungen werden, da zwischen den weit entfernten diskreten Schritten des lokalen Planers kollidierende Konfigurationen nicht entdeckt werden. Iterativ wird δ verkleinert, so dass Kollisionen von Teilbereichen des erstellten Pfades erkannt werden. Der generierte Baum zerfällt durch das Löschen der kollidierenden Zwischenstücke in mehrere Unterbäume. Der Algorithmus versucht, diese Teilstücke zu verbinden und findet so ausgehend von einer schlechten Anfangslösung iterativ oft bessere Pfade, bis schließlich

ggf. das ursprüngliche Problem mit einer kleinen Schrittweise kollisionsfrei gelöst werden kann.

5.2 Besondere Herausforderungen bei der Analyse toleranzbehafteter Bauteile

Beim Einsatz von bestehenden Pfadplanungsverfahren zur geometrieorientierten Toleranzanalyse ergeben sich einige Herausforderungen: Die in Kapitel 5.1 beschriebenen Algorithmen beschränken sich darauf, einen einzigen Montagepfad für ein Bauteil und dessen Umgebung zu berechnen. Um Aussagen treffen zu können, die statistisch aussagekräftig sind, müssen bei der Toleranzanalyse hingegen sehr viele verschiedene (z. B. simulierte) geometrische Varianten des montierten Bauteils und der Umgebung auf Fügbarkeit untersucht werden. Daher muss eine einzelne Pfadüberprüfung in kurzer Zeit abgearbeitet werden können. Bei Montageuntersuchungen treten meist Situationen auf, in denen das gefügte Bauteil nahezu keinen Bewegungsspielraum mehr besitzt. Dies resultiert in engen Durchgängen im sechsdimensionalen Suchraum, die in [32] als lange, dünne Röhren (*long, thin tube*) bezeichnet werden. Für die Lösung des Pfadplanungsproblems bei Montageaufgaben ist es daher sehr schwierig, diese Durchgänge zu finden und zu durchschreiten. Weiterhin muss nach einer begrenzten Rechenzeit eine möglichst verlässliche Aussage dahingehend getroffen werden können, ob der Montagevorgang durchführbar ist.

Die Herausforderung besteht in meiner Arbeit also zusammengefasst darin, einen Algorithmus zu entwickeln, der in kurzer Rechenzeit anspruchsvolle Pfadplanungsprobleme löst und gleichzeitig eine hohe Zuverlässigkeit aufweist.

5.3 Entwickelter Pfadplanungsansatz zur statistischen Toleranzanalyse

Um randomisierte Pfadplanungsalgorithmen trotz der aufgezeigten Schwierigkeiten anwenden zu können, wurden Anpassungen programmiert, die auf folgenden Problemeigenschaften basieren: Die fertigungsbedingten Unterschiede zwischen den Varianten desselben Bauteils sind im Vergleich zur räumlichen Ausdehnung der gesamten Montageoperation relativ klein und durch Toleranzen eingeschränkt. Da die Bauteilgeometrie sehr ähnlich ist, lässt sich schlussfolgern, dass sich auch die Montagepfade der fertigungsbedingt entstehenden Bauteilvarianten nur leicht unterscheiden. Des Weiteren kommen bei der Bauteilmontage sehr häufig bestimmte

Bewegungsarten vor, beispielsweise Translationen entlang eines Vektors oder Rotationen des montierten Bauteils. Diese eigenen Überlegungen zu den Eigenschaften des Pfadplanungsproblems für toleranzbehaftete Bauteile waren die Grundlage für die folgenden Neuentwicklungen. Sie wurden bei der Erweiterung und Anpassung der Pfadplanung zur Toleranzanalyse genutzt, wie in den folgenden Unterkapiteln beschrieben.

5.3.1 Verwenden eines Hilfspfades

Wurde für eine Variante des untersuchten Montageproblems ein Pfad gefunden, so kann dieser Pfad ggf. alle Varianten des Problems lösen. Falls nötig müssen noch kleine Anpassungen am Pfad durchgeführt werden, um Kollisionsfreiheit zu erreichen. Es sind zwar Fälle konstruierbar, bei denen kleine Geometriedefekte einen völlig neuen Montagepfad erforderlich machen, dies ist jedoch für Baugruppen ein Ausnahmefall. In [138] wurde gezeigt, wie sich die Ähnlichkeit der Pfade ausnutzen lässt:

Zunächst wird ein Montagepfad, z. B. für die Nominalbaugruppe generiert. Dies kann beispielsweise durch eine ausreichend lange Suche mit RRT-Verfahren oder durch Vorgabe durch den Anwender geschehen. Der Pfad wird so weit vereinfacht, dass er nur noch essentielle Schlüsselpositionen enthält, siehe Bild 36. Details zum eingesetzten Verfahren der Pfadvereinfachung sind in Anhang (2) beschrieben. Der auf wenige Schlüsselpositionen reduzierte Montagepfad wird als Hilfspfad bezeichnet.

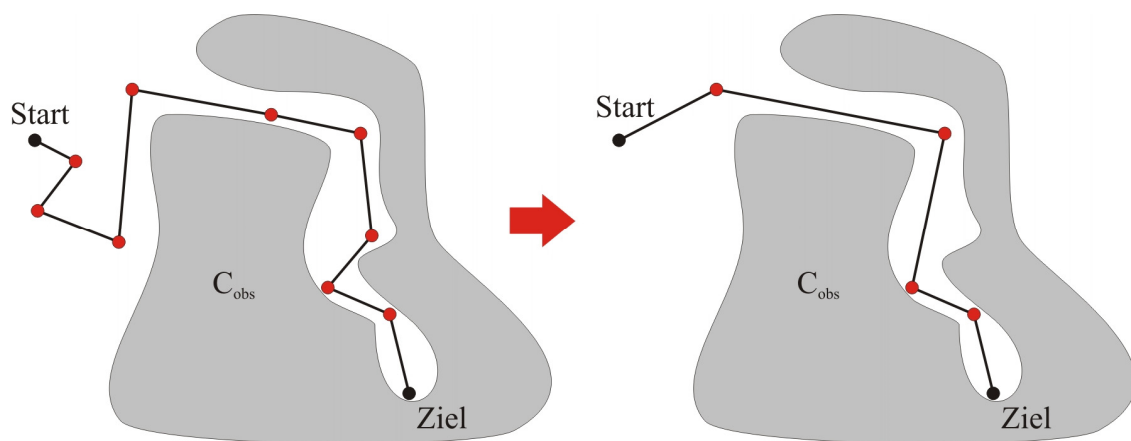


Bild 36: Reduktion eines Montagepfades auf Schlüsselpositionen

Für die Vielzahl der zu überprüfenden Bauteil- und Umgebungsvarianten wird jeweils ein neuer Montagepfad berechnet. Die Schlüsselpositionen des Hilfspfades werden als Vorgabe für die Berechnung verwendet und dienen dazu, das montierte Bauteil durch den Lösungsraum zu leiten. Die Unterteilung des Hilfspfades in zahlreiche Zwischenpositionen wurde untersucht, stellte sich jedoch als nicht sinnvoll heraus.

Durch die leicht variierende Bauteil- und Umgebungsgeometrie ändert sich bei jeder Untersuchung der Hindernisraum C_{obs} . Schlüsselpositionen des Hilfspfades können aus diesem Grund Kollisionen hervorrufen. Für kollidierende Knoten wird in der unmittelbaren Umgebung nach Alternativen gesucht, indem zufällige Positionen in der direkten Umgebung generiert werden, siehe Bild 37. Die neue Zielposition (5) wird durch Relative Positionierung wie in Kapitel 4 beschrieben berechnet.

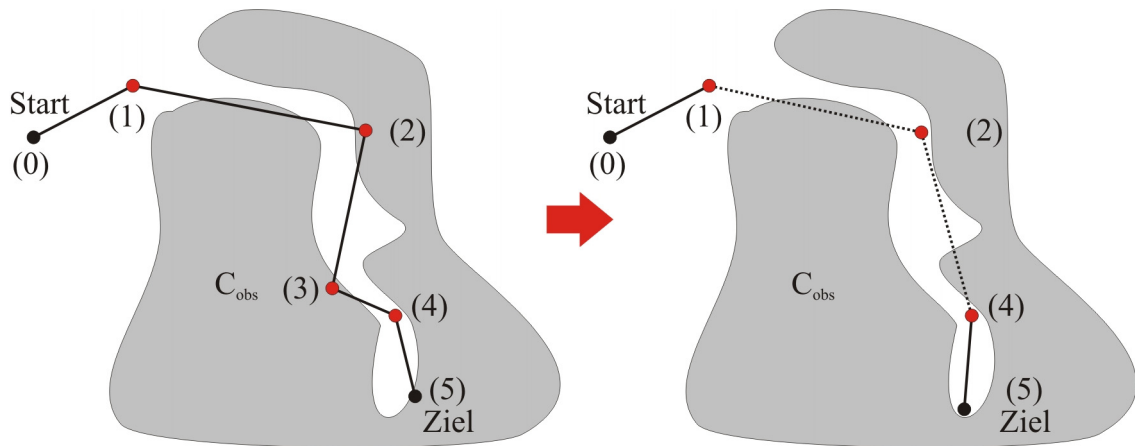


Bild 37: Anpassung kollidierender Schlüsselpositionen und der Zielposition

Gelingt es nicht, für eine Schlüsselposition eine Alternativposition zu finden, wird die jeweilige Schlüsselposition gelöscht, siehe z. B. Knoten (3) in Bild 37.

Danach beginnt die Berechnung eines zusammenhängenden Pfades von Start- zu Zielposition. Da sich das Problem in mehrere unabhängige Teilprobleme zerlegen lässt, die leichter zu lösen sind, wurde das Divide-and-Conquer-Prinzip genutzt: Für jedes verbleibende Pfadsegment wird jeweils ein eigener Pfadplanungsalgorithmus aufgerufen. Einige Segmente lassen sich direkt mit einem lokalen Planer verbinden (in Bild 37 die Abschnitte (0)-(1) und (4)-(5)). Hier muss kein Pfadplanungsprozess gestartet werden. Für die restlichen Abschnitte (im Bild gestrichelt dargestellt) muss die Pfadplanung erneut angewendet werden. Dabei ist unerheblich, welcher konkrete Pfadplanungsalgorithmus zur Lösung verwendet wird. Wurden für alle Segmente Pfade gefunden, so ist die gesamte Montage von Startposition zu Zielposition möglich. Lässt sich ein Segment nicht verbinden, wird die Bauteilkombination als „nicht montierbar“ bewertet.

5.3.2 Einschränkung des Suchraums

Wird ein Hilfspfad verwendet, so kann der Suchraum für jedes Teilsegment des Pfades eingeschränkt werden. Damit ist jedes Teilproblem erheblich leichter lösbar. Bei der Suchraumbeschränkung werden jeweils die translatorischen Ober- und Untergrenzen in

x-, y- und z-Dimension angepasst. Es hat sich als sinnvoll erwiesen, den jeweiligen Suchraum auf den Bereich zwischen den Start- und Endknoten zuzüglich eines Mindestabstands zu beschneiden. Als Mindestabstand wurde das Minimum aus 10% der Suchraumdiagonalen und dem Parameter d gewählt. Der Parameter d ist an die Größenverhältnisse der untersuchten Baugruppe anzupassen, in den durchgeführten Experimenten hat sich der Wert $d=10$ mm als praktikabel erwiesen. Beispielhaft zeigt Bild 38 in rot die angepassten Suchräume für die Abschnitte (1)-(2) und (2)-(4). Verglichen mit dem ursprünglichen Suchraum (blau dargestellt) besitzen die einzelnen angepassten Suchbereiche ein sehr viel geringeres Volumen. Für die Abschnitte wird die Pfadplanung wie beschrieben ausgeführt.

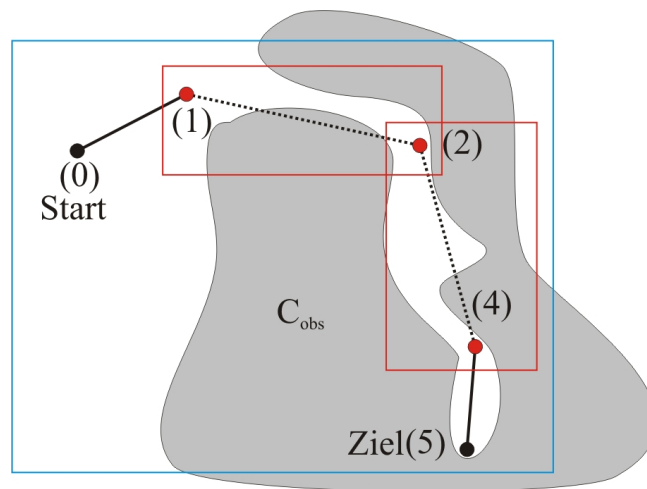


Bild 38: Einschränkung des Suchraums für zwei Teilsegmente

Wäre aufgrund von Fertigungsabweichungen für den Montagepfad ein räumlich sehr stark vom Idealpfad abweichender Pfad nötig, könnte ggf. durch die Suchraumbeschränkung eine Verbindung von Start- und Zielkonfiguration verhindert werden. Da die geometrischen Unterschiede der betrachteten Bauteile durch die definierten Toleranzen eng begrenzt sind, kann dies in der Praxis jedoch vernachlässigt werden.

5.3.3 *Angepasstes Sampling des Suchraumes*

Das beim RRT-Algorithmus (siehe [70]) Verfahren zum Erzeugen von Konfigurationen erstellt zufällige Konfigurationen im Suchraum. Für die analysierten Probleme hat es sich aus verschiedenen Gründen als zielführend herausgestellt, angepasste Samplingverfahren zu entwickeln:

- Bei der Montage von Bauteilen sind oft nur translatorische Bewegungen nötig, zusätzliche Rotationen treten nicht auf. Daher wurde ein Sampler implementiert,

der ausschließlich Translationsmatrizen erzeugt; das angegebene Montagebauteil wird nicht rotiert.

- Bei der Neuberechnung eines Teilabschnittes des Montagepfades liegen sinnvolle Zwischenpositionen meist nahe der direkten Verbindung von Start- und Zielkonfiguration (vgl. Bild 37). Ursache hierfür ist, dass der ideale Montagepfad dem angepassten meist stark ähnelt und keine völlig abweichenden Bauteilbewegungen zur erfolgreichen Montage nötig sind. Daher wurde ein Sampler programmiert, der durch sphärische lineare Interpolation eine Konfiguration auf der direkten Verbindung zwischen Start- und Endkonfiguration erzeugt. Der Sampler multipliziert diese Konfiguration mit einer Matrix, die eine kleine zufällige Verschiebung und Rotation durchführt.
- Häufig werden Bauteile auf oder zwischen achsparallelen Flächen positioniert. Dies erfordert eine Sperrung von zwei Rotationsachsen, das Bauteil rotiert beispielsweise nur um die x-Achse. Ein weiterer implementierter Sampler führt daher eine zufällige Translation durch und rotiert anschließend um die x-, y- oder z-Achse.

Die Sampler wurden so aufgebaut, dass eine gleichzeitige Verwendung mehrerer Samplerklassen möglich ist. Über die Angabe von Gewichten kann die Wahrscheinlichkeit definiert werden, mit der ein bestimmter Sampler ausgewählt wird.

5.4 Experimentelle Ergebnisse

5.4.1 Vergleich bestehender Verfahren

Um für die weiteren Arbeiten geeignete Verfahren auszuwählen, wurden zunächst einige Standard-Pfadplanungsalgorithmen auf ein typisches Montageproblem angewendet. In den Vergleich aufgenommen wurde das Multi-Query Verfahren Visibility-based Probabilistic Roadmap (PRM-Vis, siehe [89]). Ebenso wurden die in [68], [71] beschriebenen, für holonome Aufgaben am besten geeigneten Single-Query-Ansätze des Rapidly Exploring Random Tree als einseitige Variante (RRT-Con) und als bidirektionale Variante (RRT-ExtCon) verglichen. Bei den RRT-Verfahren wurde außerdem ein Vergleich zwischen linearer und logarithmischer Nachbarschaftssuche von 6D-Positionen durchgeführt. Dazu wurde die Bibliothek MPNN von A. Yershova und S. LaValle integriert (siehe [140]). Details zur linearen Nachbarschaftssuche mit Metrik M1, zur Bibliothek MPNN und der von der Bibliothek eingesetzten Metrik M2 sind in Anhang (3) beschrieben. Die MPNN-Algorithmen sind mit (*) markiert. Der

beidseitige RRT wurde als *balanced* konfiguriert. Hierbei wird immer der Suchbaum um neue Knoten erweitert, der momentan weniger Knoten besitzt.

Als erstes Problem zum Vergleich der Algorithmen wurde eine Schiene gewählt, in die ein Schlitten durch eine Öffnung eingeführt und bis ans Ende geschoben wird. Beide Bauteile liegen in ihrer Nominalgestalt vor. Die Aufgabenstellung beinhaltet einen langen, engen Tunnel im 6D-Lösungsraum mit einer Engstelle an der Schienenöffnung.

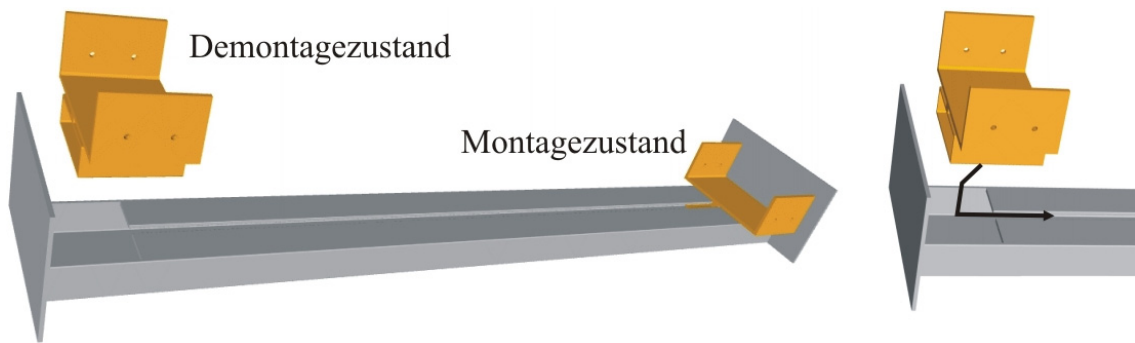


Bild 39: Montage eines Schlittens in eine Schiene (links: Start- und Zielzustand, rechts: Einführen des Schlittens in die Öffnung der Schiene)

Protokolliert wurden bei den Experimenten der Prozentsatz an erfolgreichen Montagen (*Erfolg*), die Anzahl der Knoten des aufgebauten Suchgraphen (*Knoten*), die Anzahl von Kollisionsprüfungen (Collision Detection, *CDT*) und die Rechenzeit $t(sec)$. Angegeben sind jeweils Mittelwert $\bar{}$ und Standardabweichung σ aus 50 Durchläufen des gleichen Experiments. Als Terminierungskriterium wurde eine Maximalzahl von 500.000 eingefügten Zufallsknoten für die RRT-basierten Verfahren eingestellt.

Beim Multi-Query-Verfahren PRM-Vis wurde eine Beschränkung auf maximal 2000 erfolglose in Folge generierte Zufallsknoten in der Lernphase festgelegt, um vergleichbare Rechenzeiten zu erhalten. Nach eigener Einschätzung sind PRM-basierte Multi-Query-Verfahren für toleranzbehaftete Montageaufgaben schlecht geeignet: Für jeden Lauf des PRM-Verfahrens muss die Roadmap neu aufgebaut werden, da sich der Hindernisraum durch die Geometrieabweichungen verändert. Des Weiteren geht das Verfahren bei Montageaufgaben nicht zielgerichtet vor, sondern überdeckt den Suchraum zufällig. Weil C_{free} bei Montageaufgaben einen sehr engen Tunnel im Suchraum beschreibt, ist die Wahrscheinlichkeit, durch den PRM-Algorithmus Samples in diesem Bereich zu generieren, gering.

Der translatorische Suchraum wurde für alle Experimente begrenzt auf ($2\text{ mm} < x < 5\text{ mm}$, $-1\text{ mm} < y < 105\text{ mm}$, $-100\text{ mm} < z < 750\text{ mm}$). Dies entspricht in der y - und z -Dimension den Ausmaßen der Schiene. Der rotatorische Bewegungsspielraum wurde nicht eingeschränkt.

Alle Pfadplanungsexperimente wurden auf einem PC mit Intel Core2 Quad CPU Q6600 mit 2,3 GHz, vier Rechenkernen und 3,5 GB RAM durchgeführt. Als Betriebssystem kam Windows XP SP3 zum Einsatz. Tabelle 4 zeigt die erzielten Ergebnisse der Experimente.

Tabelle 4: Vergleich der Basisalgorithmen zur Pfadplanung am Beispiel Schlitten-Schiene mit nominalen Bauteilen

	Erfolg	Knoten: \bar{O} , σ	CDT: \bar{O} , σ	t(sec): \bar{O} , σ
PRM-Vis	0%	68, 17	44.086.890, 18402.982	1.264, 534
RRT-Con	62%	1.430, 466	294.516, 190.001	2.088, 1.263
RRT-Con*	42%	5.446, 3.105	359.577, 205.414	1.992, 1.037
RRT-ExtCon	0%	7.240, 191	511.346, 88	1.721, 55
RRT-ExtCon*	0%	8.359, 326	511.891, 138	1.213, 30

Die experimentellen Ergebnisse bestätigen die Einschätzung, dass PRM-basierte Multi-Query-Verfahren für die Montagesimulation nicht geeignet sind. Der Algorithmus PRM-Vis war nicht in der Lage, die Montagekonfiguration mit neuen Knoten zu verbinden, da der "Montage-Tunnel" im Vergleich zum gesamten Suchraum verschwindend klein ist. Die Visualisierung der aufgebauten Suchgraphen in Bild 40 zeigt jeweils die Translationen des Mittelpunkts des bewegten Schlittens, Bild 40(1) exemplarisch einen von PRM-Vis generierten Suchgraphen, der ausschließlich die leicht erreichbare Umgebung der Demontageposition überdeckt. Im Montagetunnel wurden durch das zufällige Sampling keine Knoten erzeugt.

Wie in [32] beschrieben, verbrauchen auch die bidirektionalen RRT-Verfahren einen großen Anteil der Rechenzeit damit, den kollisionsfreien Raum zu erkunden (siehe Bild 40(2)). Durch die erfolglosen Verbindungsversuche der Bäume verlangsamt sich das Voranschreiten des von der Montageposition ausgehenden unteren Suchbaumes zusätzlich.

Da das einseitige RRT-Verfahren vom Montagezustand aus beginnt, den Suchraum zu erkunden, gelingt es hier häufig, den Schlitten zu montieren, siehe Bild 40(3).

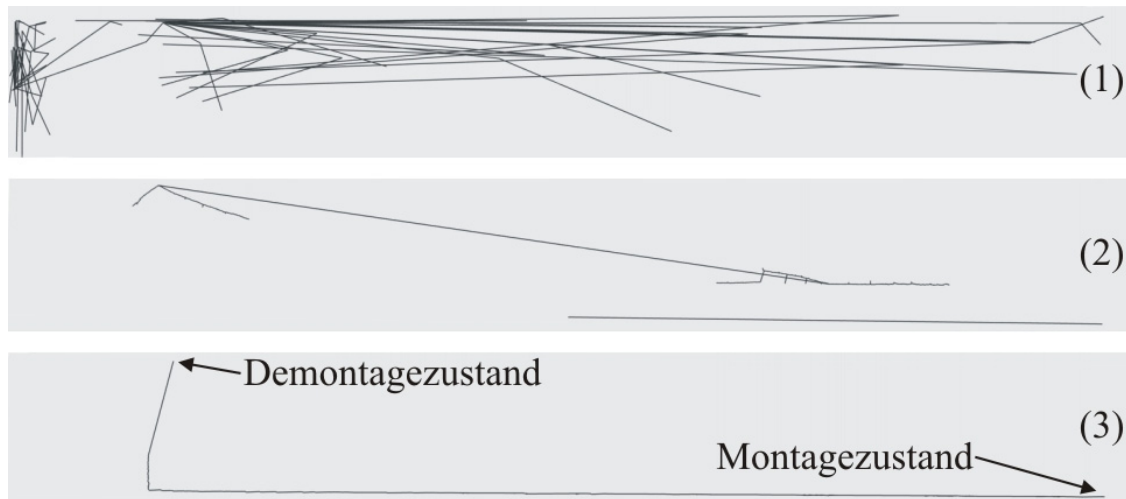


Bild 40: Typische Suchgraphen der Verfahren PRM-Vis (1), RRT-ExtCon(2) und RRT-Con(3)

Für die Konstellation Schlitten-Schiene wurden nichtideale Varianten beider Bauteile erzeugt. Diese sind zwar montierbar, allerdings ist durch die Geometrieabweichungen C_{free} weiter eingeschränkt. Die Montageposition wurde durch Relative Positionierung bestimmt. Tabelle 5 zeigt die experimentellen Ergebnisse für die Montagesimulation der nichtidealen Bauteilvarianten Schiene_9 und Schlitten_9.

Tabelle 5: Ergebnisse der Pfadplanung für das Problem Schlitten-Schiene (nichtideale Baugruppe)

	Erfolg	Knoten: \emptyset, σ	CDT: \emptyset, σ	t(sec): \emptyset, σ
RRT-Con	0%	447, 61	511.103, 1.271	1.564, 258
RRT-Con*	0%	579, 101	511.228, 1.315	1.105, 180
RRT-ExtCon	0%	516, 56	508.968, 29	668, 27
RRT-ExtCon*	0%	516, 63	508.970, 27	646, 24

Aufgrund der geometrischen Abweichungen war keines der verglichenen Verfahren mehr in der Lage, eine erfolgreiche Montage durchzuführen. Es wurden Graphen mit erheblich weniger Knoten aufgebaut, der Faktor betrug für die einseitigen RRT-Algorithmen (RRT-Con und RRT-Con*) 3 bis 11, bei den beidseitigen Algorithmen 14 bis 16. Bei der Visualisierung der generierten Graphen (Bild 41) ist zu erkennen, dass die Suchbäume im Montagetunnel deutlich kürzere Montagestrecken erkunden, als bei der Simulation der Nominalbauteile (Bild 40).



Bild 41: Suchgraphen von RRT-ExtCon (1) und RRT-Con (2) bei nichtidealen Varianten von Schlitten und Schiene

Als zweite Pfadplanungsaufgabe wurde das sog. „Alpha-Puzzle“ verwendet. Das Alpha-Puzzle ist ein bekanntes geometrisches Problem zum Vergleich von Pfadplanungsalgorithmen, siehe [139]. Zwei röhrenförmige Teile, die die Form des griechischen Buchstabens Alpha besitzen, sind ineinander verschlungen (Bild 42). Jedes der beiden Bauteile besteht aus 1.008 Dreiecken. Ziel des Puzzles ist es, die Teile voneinander zu trennen. Der translatorische Suchraum des Problems wurde festgelegt auf $(-20 \text{ mm} < x < 5 \text{ mm}, 0 \text{ mm} < y < 40 \text{ mm}, -10 \text{ mm} < z < 40 \text{ mm})$, die Rotationen wurden nicht eingeschränkt.

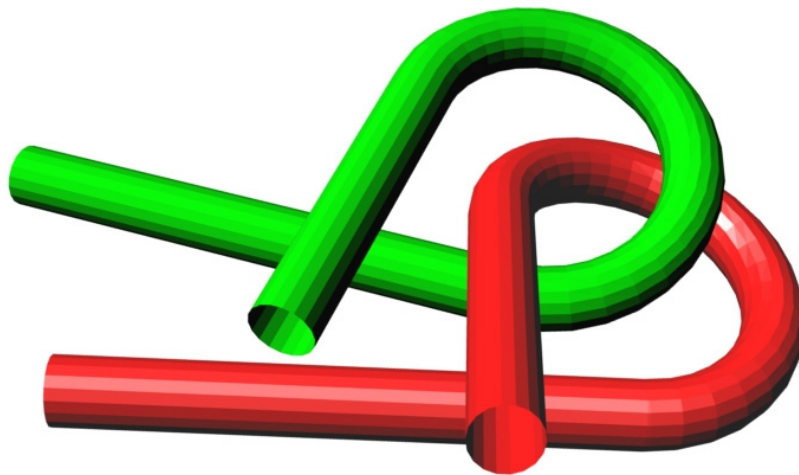


Bild 42: Alpha-Puzzle mit der Skalierung 1.1 (rot: Umgebung, grün: bewegtes Teil)

Für das Problem existieren unterschiedliche Schwierigkeitsgrade, bei denen das unbewegte Bauteil (rot dargestellt) in z-Richtung skaliert wurde. Durch die Skalierung verändert sich die Spaltbreite zwischen den roten Röhrenenden des unbewegten Teils und damit der Schwierigkeitsgrad des Problems. Den größten Abstand der durchgeführten Experimente besitzt AlphaPuzzle_1.2, den kleinsten AlphaPuzzle_1.0. Eine Demontage ist ab AlphaPuzzle_1.1 nicht mehr durch

einfache Translation möglich. Nur durch eine komplexe Folge von Verschiebungen und Rotationen können die Teile getrennt werden, siehe Bild 43.

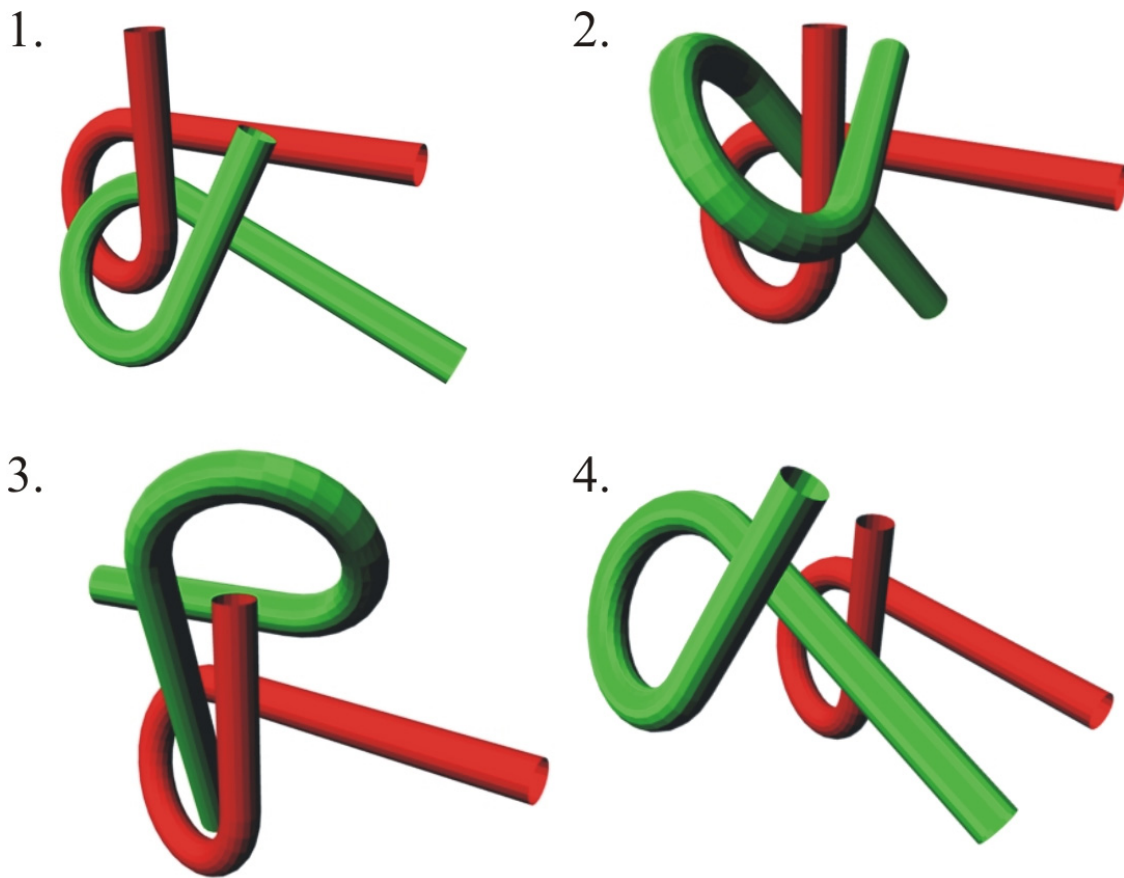


Bild 43: Lösung des Pfadplanungsproblems AlphaPuzzle_1.0

Die erzielten Ergebnisse der vorgestellten Pfadplanungsalgorithmen sind in Tabelle 6, Tabelle 7 und Tabelle 8 für die Schwierigkeitsstufen 1.2, 1.1 und 1.0 zusammengefasst.

Tabelle 6: Vergleich der Basisalgorithmen (AlphaPuzzle_1.2)

	Erfolg	Knoten: \emptyset , σ	CDT: \emptyset , σ	t(sec): \emptyset , σ
PRM-Vis	0%	191, 29	32.662.755, 11.450.910	2.632, 1.109
RRT-Con	100%	1.224, 792	55.701, 45.785	22, 26
RRT-Con*	100%	1.613, 1.340	60.172, 49.540	10, 9
RRT-ExtCon	88%	21.532, 7.559	221.558, 157.310	761, 784
RRT-ExtCon*	34%	99.144, 43.139	431.293, 186.434	130, 60

Tabelle 7: Vergleich der Basisalgorithmen (AlphaPuzzle_1.1)

	Erfolg	Knoten: \emptyset, σ	CDT: \emptyset, σ	t(sec): \emptyset, σ
RRT-Con	94%	3.253, 1.419	228.746, 147.021	166, 165
RRT-Con*	94%	17.631, 16.933	629.801, 544.390	108, 104
RRT-ExtCon	0%	33.102, 1.019	511.150, 378	2.284, 84
RRT-ExtCon*	0%	121.922, 2.512	544.252, 1.432	171, 4

Tabelle 8: Vergleich der Basisalgorithmen (AlphaPuzzle_1.0)

	Erfolg	Knoten: \emptyset, σ	CDT: \emptyset, σ	t(sec): \emptyset, σ
RRT-Con	70%	4.295, 1.880	325.280, 192.556	295, 248
RRT-Con*	82%	29.978, 19.866	1.032.201, 635.725	190, 128
RRT-ExtCon	0%	32.732, 1.240	511.070, 400	2.223, 119
RRT-ExtCon*	0%	121.441, 2.913	544.534, 1.491	171, 3

Auch hier zeigte sich das Verfahren RRT-Con den anderen Verfahren gegenüber überlegen. Das PRM-Verfahren konnte selbst bei der einfachsten Variante des Problems keine Lösung finden. Bei den Varianten 1.1 und 1.0 des Problems, die eine komplexe Folge von Translationen und Rotationen erfordern, war das beidseitige Verfahren nicht in der Lage, eine Lösung zu finden. Mit RRT-Con wurde meist nach relativ kurzer Rechenzeit eine Lösung generiert, der Algorithmus war auch bei der anspruchsvollen Variante 1.0 teilweise erfolgreich.

Des Weiteren hatte die Anwendung der Nearest-Neighbor Bibliothek sehr positiven Einfluss auf die Rechenzeit. Durch Einsatz von Metrik M2 wurden jedoch erheblich größere Suchgraphen aufgebaut und die Kollisionserkennung wurde öfter aufgerufen.

Aufgrund der im Algorithmenvergleich ermittelten Daten wurden in den folgenden Experimenten nur noch RRT-Verfahren eingesetzt. Das Probabilistic Roadmap-Verfahren war bei den Experimenten nicht in der Lage, die engen Suchraumpassagen von Montageaufgaben zu durchschreiten und erkundete hauptsächlich den freien Raum.

Einseitige RRTs haben sich insbesondere für Problemstellungen als geeignet erwiesen, in denen das bewegte Bauteils nahe der Demontageposition einen großen Bewegungsspielraum zur Verfügung hat und der Demontagepfad eng eingeschränkt ist. Das einseitige RRT-Verfahren erkundete vergleichsweise zuverlässig den „Montage-Tunnel“.

Keines der Basisverfahren der Pfadplanung war in der Lage, die Montage der nichtidealen Bauteilvarianten des Problems Schlitten-Schiene zu simulieren. Somit lässt sich aus dem Vergleich der Algorithmen das Fazit ziehen, dass eine Anpassung und Erweiterung der Basisalgorithmen zwingend notwendig ist, um stochastische Montagesimulationen durchzuführen.

5.4.2 Einbinden eines Hilfspfades

Um den Einfluss von Hilfspositionen auf die Montageuntersuchung Schlitten-Schiene zu untersuchen, wurde ein Hilfspfad hinzugefügt, der aus einem erfolgreichen Simulationsablauf der Idealteile (vgl. Kapitel 5.4.1) stammt. Der Hilfspfad besitzt neben Start- und Zielposition zwei zusätzliche Zwischenpositionen (siehe Bild 44).



Bild 44: Schematische Darstellung des Hilfspfades aus vier Positionen (rot)

Dadurch zerfällt das Montageproblem in drei Zwischenstücke, die unabhängig voneinander bearbeitet werden. Angewendet wurden die RRT-Verfahren auf die in Kapitel 5.4.1 eingesetzte nichtideale Variante der Schiene in Kombination mit einem nichtidealen Schlitten.

Tabelle 9: Vergleich der Pfadplanungsalgorithmen bei Verwendung eines Hilfspfades (Problem: Schlitten-Schiene, nichtideale Bauteile)

	Erfolg	Knoten: \emptyset, σ	CDT: \emptyset, σ	t(sec): \emptyset, σ
RRT-Con	10%	1.264, 5.813	645.512, 1.103.158	3.960, 926
RRT-Con*	68%	1.840, 9.829	1.618.55, 9.028.352	2.805, 1.267
RRT-ExtCon	0%	434, 58	501.768, 448	852, 102
RRT-ExtCon*	0%	471, 60	501.805, 414	818, 114

Die Erfolgswahrscheinlichkeit der einseitigen RRT-Verfahren stieg durch Verwendung der Hilfspositionen von 0% (Basisalgorithmen) auf 10% bzw. 68%. Es konnten erheblich größere Suchbäume generiert werden, allerdings stieg die benötigte Rechenzeit ebenfalls an.

Die beidseitigen Verfahren waren trotz der Hilfeposition nicht in der Lage, einen zusammenhängenden Pfad zu finden.

In diesem Experiment zeigte die Metrik M_2 bei Einsatz des Verfahrens RRT-Con eine höhere Erfolgswahrscheinlichkeit als Metrik M_1 .

5.4.3 Hilfefad mit Suchraumanpassung

Zusätzlich zur Angabe von Hilfsposition wurde im folgenden Experiment der Suchraum, wie in Kapitel 5.3.2 beschrieben, automatisch beschränkt. Das Suchraumvolumen sinkt mit dieser Anpassung auf 13% des ursprünglichen Volumens.

Tabelle 10: Vergleich der Pfadplanungsalgorithmen bei Verwendung eines Hilfefads und automatischer Suchraumanpassung (Problem: Schlitten-Schiene)

	Erfolg	Knoten: \emptyset, σ	CDT: \emptyset, σ	t(sec): \emptyset, σ
RRT-Con	94%	1.015, 535	184.344, 149.007	2.236, 1.578
RRT-Con*	100%	536, 295	33.870, 17.960	614, 199
RRT-ExtCon	6%	828, 112	483.793, 79.178	2.558, 383
RRT-ExtCon*	0%	4.118, 331	505.087, 378	2.303, 74

Wie in Tabelle 10 zu sehen ist, konnte durch die automatische Suchraumanpassung die Erfolgswahrscheinlichkeit des RRT-Con-Verfahrens stark verbessert werden. Gleichzeitig ist ein Absinken der Rechenzeit zu vermerken. Das Verfahren RRT-Con* war in der Lage, für alle 50 Durchläufe Lösungen zu generieren.

Durch die Erweiterungen war der bidirektionale Algorithmus RRT-ExtCon befähigt, das Pfadplanungsproblem für nichtideale Bauteile in 6% der Durchläufe zu lösen.

5.4.4 Verwendung angepasster Samplingverfahren

Beim folgenden Experiment wurde ebenfalls die Problemstellung Schlitten-Schiene für die nichtideale Baugruppe untersucht. Jedoch wurden neben dem Standard-Sampling-Verfahren zusätzlich die in 5.4.4 beschriebenen Sampler verwendet. Tabelle 11 zeigt die verwendete Gewichtung, die aufgrund der Problemstellung gewählt wurde. Die Montageaufgabe besteht vornehmlich aus translatorischen Bewegungen des Schlittens.

Tabelle 11: Gewichtung der Samplingverfahren

Zufällige Samples im Suchraum (sampleGenerator_Random)	14%
Nur Translatorische Bewegungen (sampleGenerator_Translation)	55%
Samples nahe der direkten Verbindung (sampleGenerator_BetweenPos)	30%
Nur Rotationen um eine Suchraumachse (sampleGenerator_AxisRotation)	1%

Es wurden keine Hilfepositionen angegeben, der Suchraum wurde nicht eingeschränkt. Die Ergebnisse sind in Tabelle 12 zusammengefasst.

Tabelle 12: Verwendung angepasster Samplingverfahren (Problem: Schlitten-Schiene, nichtideale Baugruppe)

	Erfolg	Knoten: \emptyset , σ	CDT: \emptyset , σ	t(sec): \emptyset , σ
RRT-Con	98%	1.163, 591	196.781, 591	1.503, 702
RRT-Con*	88%	2.144, 1.372	245.695, 169.016	1.469, 668
RRT-ExtCon	0%	7.865, 213	511.923, 102	1.263, 40
RRT-ExtCon*	0%	8.163, 218	512.060, 108	839, 24

Das einseitige RRT-Verfahren profitierte enorm von den angepassten Samplingverfahren: Die Erfolgswahrscheinlichkeit stieg im Vergleich zum Basisalgorithmus (siehe 5.4.1) von 0% auf 98% bzw. 88%. Der Algorithmus erzeugte für die nichtideale Baugruppe erheblich größere Suchbäume als das Basis-Verfahren (vgl. Tabelle 5).

Das bidirektionale Pfadplanungsverfahren war nicht in der Lage, das Montageproblem zu lösen. Die aufgebauten Suchbäume ähneln stark den Suchbäumen des Basisalgorithmus (siehe Bild 40).

5.4.5 Kombination aller entwickelten Verfahren

Abschließend wurden alle entwickelten Verfahren zur Steigerung der Rechengeschwindigkeit und Zuverlässigkeit gemeinsam zur Lösung von Pfadplanungsproblemen eingesetzt. Neben der Vorgabe eines Hilfspfades (siehe Kapitel 5.4.2) kamen automatische Suchraumanpassung (Kapitel 5.4.3) und angepasste Samplingverfahren (5.4.4) zur Anwendung.

Es wurde die bereits beschriebene Montage von Schlitten und Schiene simuliert. Untersucht wurden zehn nichtideale Schlitten und zehn nichtideale Schienen. Zunächst wurden die finalen Montagepositionen der 100 möglichen Kombinationen durch Relative Positionierung bestimmt. Anschließend wurde als Hilfspfad ein aus Experiment 5.4.1 stammender Pfad der Nominalteile verwendet, der Suchraum wurde automatisch eingeschränkt. Zur Abtastung des Suchraums wurden die in 5.3.3 beschriebenen Samplingverfahren mit der Gewichtung aus Tabelle 11 eingesetzt. Jede Montage der einhundert möglichen Teilkombinationen wurde jeweils 10-mal simuliert.

Tabelle 13: Pfadplanungsergebnisse bei Integration aller entwickelter Erweiterungen (Problem: Schlitten-Schiene, nichtideale Bauteilvarianten)

	Erfolg	Knoten: \emptyset, σ	CDT: \emptyset, σ	t(sec): \emptyset, σ
RRT-Con	100%	271, 558	14.479, 26.786	453, 300
RRT-Con*	100%	278, 681	14.275, 36.081	430, 273
RRT-ExtCon	98%	206, 822	21.176, 57.517	403, 345
RRT-ExtCon*	98%	398, 1.367	29.283, 83.808	403, 317

Das Experiment ergab, dass alle untersuchten abweichungsbehafteten Bauteilvarianten miteinander kombinierbar sind und die Montage stets durchführbar ist. Der einseitige RRT-Algorithmus erzeugte für 100% der Kombinationen kollisionsfreie Lösungen und arbeitete damit zuverlässiger als das beidseitige Verfahren, das Montagepfade für 98% der Probleme generieren konnte. Die Rechenzeit reduzierte sich im Vergleich zu den nichtoptimierten Standard-Verfahren (siehe Kapitel 5.4.1) auf durchschnittlich 403 s bis 453 s. Es wurden kleinere Suchbäume als bei den Basisalgorithmen aufgebaut und die Kollisionserkennung wurde seltener aufgerufen.

Als weiteres Experiment zur Überprüfung des entwickelten Ansatzes wurde das Pfadplanungsproblem `AlphaPuzzle_1.0` untersucht. Eine Lösung der Variante `AlphaPuzzle_1.1` (siehe Kapitel 5.4.1) wurde als Hilfspfad verwendet. Vergleichbar ist diese Problemstellung mit der Analyse von simulierten nichtidealen Bauteilen, wobei die Umgebungsvariante 1.0 einem Teil am unteren Spezifikationslimit mit sehr engem Spaltmaß entspricht.

Tabelle 14: Pfadplanungsergebnisse bei Integration aller entwickelter Erweiterungen (Problem: AlphaPuzzle_1.0)

	Erfolg	Knoten: \emptyset, σ	CDT: \emptyset, σ	t(sec): \emptyset, σ
RRT-Con	96%	7.651, 7.031	203.453, 178.827	299, 492
RRT-Con*	100%	379, 317	11.443, 7.460	5,59, 1,32
RRT-ExtCon	26%	50.101, 7.824	487.510, 84.036	2.675, 717
RRT-ExtCon*	100%	7.711, 2.665	37.675, 13.041	8,95, 1,83

Durch die Verwendung des Hilfspfades konnte das Problem vom einseitigen RRT-Verfahren in 96% bzw. 100% der Durchläufe gelöst werden. Das beidseitige RRT-Verfahren fand mit Metrik M_1 in 26% der Simulationen eine Lösung, mit Metrik M_2 in 100%. Erneut zeigte sich, dass durch die Spezialisierung der Pfadplanungsalgorithmen

auf Toleranzuntersuchungen eine höhere Zuverlässigkeit bei gleichzeitig kürzeren Rechenzeiten realisierbar ist.

5.5 Pfadplanung: Zusammenfassung und Ausblick

Die Herausforderung bei der Montagesimulation nichtidealer Bauteilvarianten besteht darin, in einem hochdimensionalen Suchraum einen Pfad zu finden, obwohl für das jeweils bewegte Teil nur ein geringer Bewegungsspielraum zur Verfügung steht. Zusätzlich muss die Simulation möglichst schnell ablaufen, da statistische Untersuchungen der Baugruppen erwünscht sind und daher viele Baugruppenvarianten zu untersuchen sind. Das Berechnungsergebnis muss außerdem möglichst zuverlässig sein, auch wenn aufgrund der Suchraumdimensionalität nur probabilistische Verfahren einsetzbar sind.

Im vorangegangenen Text wurde gezeigt, wie sich automatische Pfadplanungsalgorithmen zur Analyse abweichungsbehafteter Baugruppen erweitern lassen, um so den genannten Anforderungen zu genügen. Genutzt wurde bei der Anpassung und Erweiterung der Basisalgorithmen die Tatsache, dass sich der Montagepfad simulierter, nichtidealer Bauteilvarianten nur geringfügig vom Pfad der Nominalteile unterscheidet. Daher lässt sich der einmal berechnete Idealpfad der Nominalbauteile als zusätzliche Eingabe bei den folgenden Pfadsuchen nutzen. Durch automatische Einschränkung des Suchraums und spezielle Samplingverfahren konnte die Rechenzeit und Zuverlässigkeit weiter verbessert werden. Die Rechenzeit sank (je nach Montageproblem) auf wenige Sekunden bzw. Minuten. Für den Einsatzzweck der Toleranzanalyse ist jedoch die Zuverlässigkeit der Algorithmen wichtiger, die sich im Vergleich zu den Basisalgorithmen ebenfalls stark steigern ließ. Daher kann man das vorgeschlagene Verfahren bereits für eine mittlere Anzahl (im Bereich von 10^3) von Bauteilvarianten praktisch einsetzen. Als zuverlässigstes Verfahren für Demontageuntersuchungen hat sich in den Experimenten das einseitige RRT-Verfahren RRT-Con herausgestellt. Durch den Einsatz der Bibliothek MPNN zur schnellen Nachbarschaftssuche ließen sich in vielen Experimenten deutliche Geschwindigkeitssteigerungen erzielen. Bei den eingesetzten Metriken ließ sich kein klarer Vorteil einer Metrik herausstellen. Je nach Experiment lieferte Metrik M_1 oder M_2 bessere Ergebnisse.

Die in [115] beschriebene Kombination aus FE-Simulation und Latin Hypercube Sampling ist bereits in der Lage, realistische Geometrie für Tiefziehprozesse als

Eingangsdaten zu erzeugen. Nach Ablauf der Pfadplanung für eine statistisch ausreichende Menge von nichtidealen Bauteilen erhält der Anwender die Information, ob alle Teilekombinationen ffügbar sind, bzw. für welchen Prozentsatz kein Pfad generiert werden konnte. Für die nicht montierbaren Varianten kann ausgegeben werden, welcher Abschnitt des Montagevorgangs nicht kollisionsfrei durchführbar ist.

Neben der Überprüfung von Montage- bzw. Demontagevorgängen ließe sich das entwickelte Pfadplanungsverfahren zukünftig auch zur Überprüfung von nichtidealen Bauteilen verwenden, die sich im Betrieb bewegen. Die Pfadplanung kann berechnen, ob alle geometrischen Varianten in der Lage sind, die beschriebene Bewegung durchzuführen. Kräfte und Verformungen werden momentan bei der Pfadplanung nicht berücksichtigt. In [122] werden alternative Techniken vorgestellt, um statistisch die Kinematik von tolerierten Baugruppen zu analysieren.

Wie bei der Relativen Positionierung wurde bei den beschriebenen Pfadplanungsverfahren angenommen, dass die Bauteile vollkommen starr (rigid) sind. Diese Annahme ist für viele Einsatzbereiche zulässig und sinnvoll. Für die Analyse von nachgiebigen Baugruppen, beispielsweise großflächigen Kunststoffteilen oder Blechen, ist diese Modellierung allerdings nicht ausreichend. Hier müssen in Zukunft Erweiterungen entwickelt werden, die die Nachgiebigkeit des Materials von montiertem Bauteil und Umgebung sowie die wirkenden Kräfte mit einbeziehen. Auf dem Gebiet der Pfadplanung existieren hierzu noch keine Verfahren, die in der Lage sind, Ergebnisse mit der erforderlichen Genauigkeit in annehmbarer Rechenzeit zu liefern.

6 Analyseverfahren zur Auswertung von Abweichungen

Nach dem virtuellen Zusammenbau der toleranzbehafteten Baugruppen können die entstandenen Daten ausgewertet werden. Neben der Durchführung von Messungen, z. B. von Abständen, Maximalhöhe etc. lassen sich auch komplexere Analyseverfahren einsetzen. Exemplarisch werden in den folgenden Kapiteln implementierte Techniken zur visuellen und haptischen Analyse einzelner Baugruppenvarianten und zur statistischen Auswertung der Simulationsergebnisse beschrieben. Gegliedert ist das Kapitel in Verfahren zur Analyse einzelner Baugruppen (Abschnitt 6.1 und 6.2) und Verfahren zur statistischen Analyse mehrerer Baugruppen (Abschnitt 6.3).

6.1 Graphische Analyse einzelner Baugruppen durch

Abstandsmessungen

In [64] wurden bereits Visualisierungsverfahren zur Analyse abweichungsbehafteter Geometrie vorgestellt, beispielsweise Farbcodierung, Isolinien und Glyphendarstellung. Visualisiert wurde bei den genannten Verfahren der euklidische Abstand von Punkten auf der Ist-Geometrie zur Oberfläche der Soll-Geometrie.

Durch den virtuellen Zusammenbau nichtidealer Baugruppen bei der Relativen Positionierung entstehen zwischen den Einzelteilen Abstände, siehe Kapitel 4. Da die kombinierten Teile keine perfekt ebenen bzw. rechtwinkligen Flächen besitzen, liegen sie nur noch in einzelnen Flächenbereichen aufeinander auf. Zusätzlich kommt es zu Verkipnungen der Teile zueinander. Dass die Wahrnehmung von Spalt- und Versatzverläufen bei realistischer graphischer Darstellung vielen Parametern (z. B. Texturierung, Farbgebung, Stereomodus) unterliegt und personenabhängig sehr unterschiedlich beurteilt wird, wurde bereits durch Wickman et al. in [132] gezeigt. Ergänzend wurden daher Verfahren entwickelt, die speziell Abstände zwischen nichtidealen Bauteilen, sowie entstehende Spaltverläufe vermessen und visualisieren, siehe [93] und [92]. Im folgenden Text werden die hierzu selbst entwickelten Analyseverfahren beschrieben.

6.1.1 Definition von Abstandsmessungen

Die einfachste Form der Vermessung von Abständen zwischen Bauteil A und Bauteil B stellt die Zweipunktmessung dar. Zu diesem Zweck wurde ein Analysewerkzeug geschaffen, das teilautomatisch Messungen durchführt. Realisiert wurde dies dadurch, dass der Anwender mit dem Mauszeiger in 2D eine Position auswählt. Zunächst wird

aus der 2D-Position die 3D-Position des gewählten Punktes auf der Oberfläche von Bauteil A berechnet. Um die Auswahl von Bauteilkanten zu erleichtern, wird dieser Punkt um einen kleinen Offset-Vektor in Richtung des Betrachters verschoben. Zum verschobenen 3D-Punkt wird der nächste Punkt auf der Netzoberfläche von A bestimmt. Durch diese Vorgehensweise wird ein „Einrasten“ des Mauszeigers an konvexen Objekten bewirkt. Der Anwender kann somit in drei Dimensionen einen Punkt auf Bauteil A definieren.

Nach Auswahl des Aufpunktes der Messung wird automatisch der nächstgelegene Punkt auf der Oberfläche des gegenüberliegenden Dreiecksnetzes von Bauteil B gefunden. Die definierte Messung wird als Pfeildarstellung mit eingeblendetem Abstandswert in mm gezeigt, siehe Bild 45.

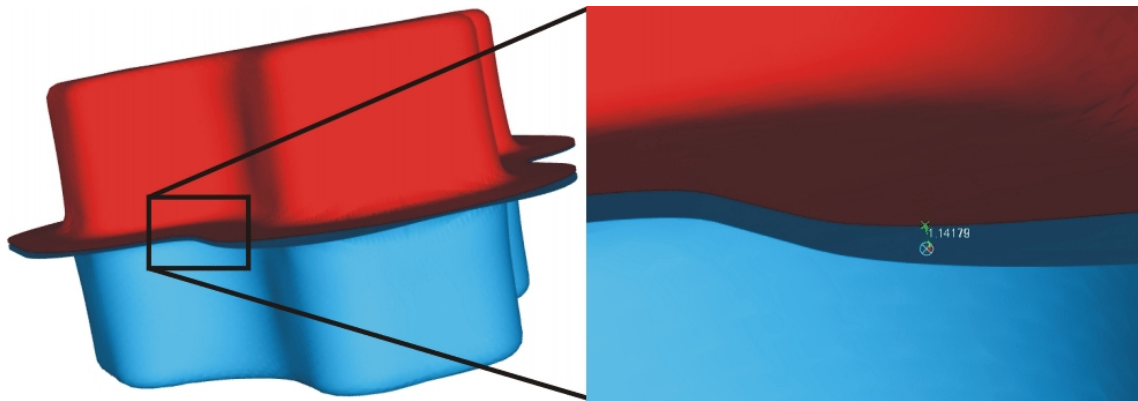


Bild 45: Zweipunktmessung im Flanschbereich zweier positionierter Tiefziehteile

Zur Beurteilung des gesamten Abstandsverlaufs zwischen Bauteilen ist die manuelle Definition aller dazu notwendigen Messpunkte zu zeitaufwändig. Daher wurde ein Verfahren implementiert, das, ausgehend von einer einzigen Zweipunktmessung, zahlreiche weitere Messungen automatisch durchführt.

Ziel der automatischen Messungserstellung ist es, zusätzliche Messungen zu definieren, die der vom Benutzer vorgegebenen Startmessung ähneln. Zwei Vergleichskriterien wurden dazu ausgewählt.

Zum einen kann für den euklidischen Abstand zwischen Teil A und B eine Ober- und Untergrenze angegeben werden. Die Grenzwerte werden relativ ausgedrückt, beispielsweise werden bei einer Startmessung von 20 mm und Einstellung der Grenzen 0,8 und 1,1 nur Abstände zwischen 16 mm und 22 mm betrachtet.

Zum anderen lässt sich definieren, wie weit die Oberflächennormalen zweier nebeneinander liegenden Messungen abweichen dürfen. Auf diese Weise lassen sich Bereiche mit kontinuierlich gekrümmten Oberflächen selektieren. Knickt das

Dreiecksnetz an einer Stelle stark ab, wird dies als Ende der zu vermessenden Fläche interpretiert.

Die automatisch generierten Zweipunktmessungen werden als Linien dargestellt, die je nach euklidischem Abstand farblich codiert sind. Zum Einfärben wurde der HSV-Farbraum verwendet, aufeinander folgen blau – türkis – gelb – orange – rot, wobei die kleinste Abstandsmessung blau eingefärbt wird. Ein Überblick über die Werteverteilung der Einzelmessungen kann durch ein Histogramm gewonnen werden.

6.1.2 Auswertung am Beispiel der Napf-Baugruppe

Der Einsatz des Visualisierungsverfahrens soll nun für verschiedene Zielsetzungen am Beispiel der bereits bekannten Napf-Baugruppe demonstriert werden.

In Kapitel 4.5.5 wurde die Positionierung nichtidealer Kreuzziehteile auf einer Vorrichtung beschrieben. Nun kann dargestellt werden, welchen Abstand das untere Bauteil der Baugruppe zur Positionierungsvorrichtung besitzt und in welchen Bereichen der Napfboden auf der Vorrichtung aufliegt. Dazu wird vom Benutzer eine Startmessung von einem beliebigen Punkt des Napfbodens zur Vorrichtung angelegt. Um für die automatische Messpunktauswahl nur die Bodenfläche des Napfes zu selektieren, wird die zulässige Normalenabweichung limitiert.

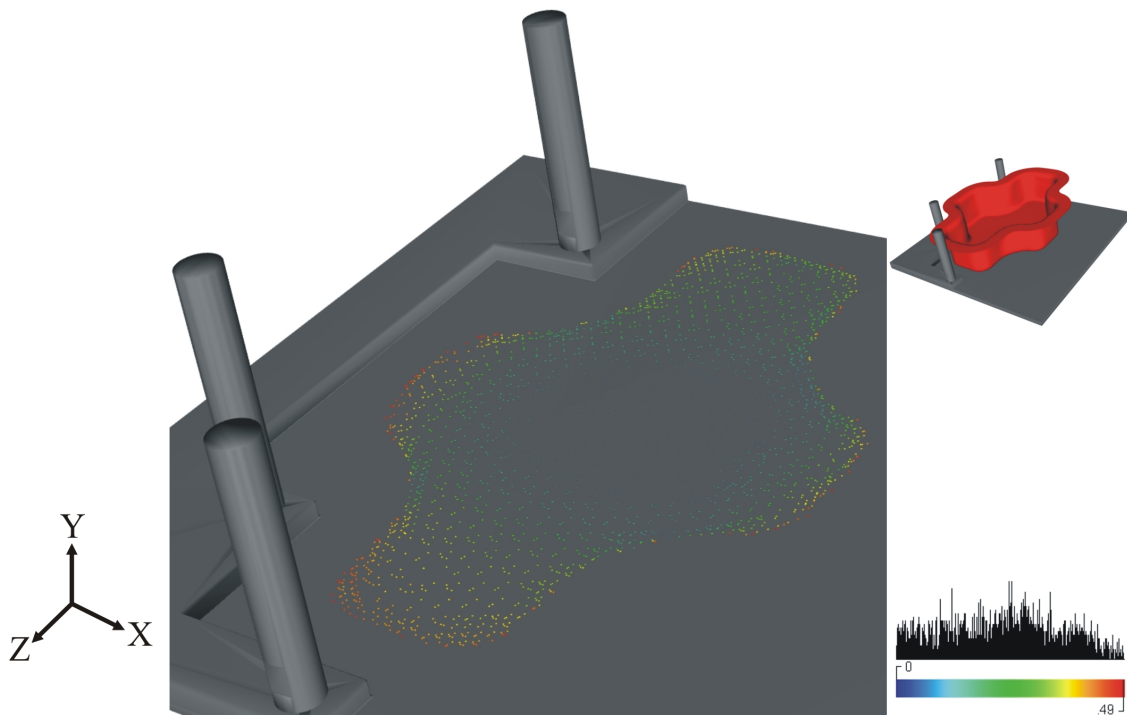


Bild 46: Abstandsmessung zwischen Kreuzziehteil und Positioniervorrichtung

Durch die so entstehende Visualisierung (siehe Bild 46) wird ersichtlich, dass das untere Kreuzziehteil vor allem im Mittelbereich des Bodens aufliegt. Die Auflagefläche ist

jedoch nicht perfekt mittig platziert, sondern leicht in negative z-Richtung und positive x-Richtung verschoben. Die größten Abstände zwischen Napfboden und Vorrichtung treten in positiver z-Richtung am Randbereich des Bodens auf, erkennbar an den gelben und roten Bereichen im unteren Teil von Bild 46. Rechts im Bild dargestellt ist das Histogramm der Messung, der Messbereich beträgt 0 mm bis 0,49 mm.

Ein weiteres Einsatzszenario der Visualisierung stellt die Untersuchung der Auflage der Näpfe im Flanschbereich dar. Dies kann nach der Positionierung des oberen Kreuzziehteils durchgeführt werden. Die Definition geschieht analog zur vorhergehenden Beschreibung durch eine Startmessung im Flanschbereich.

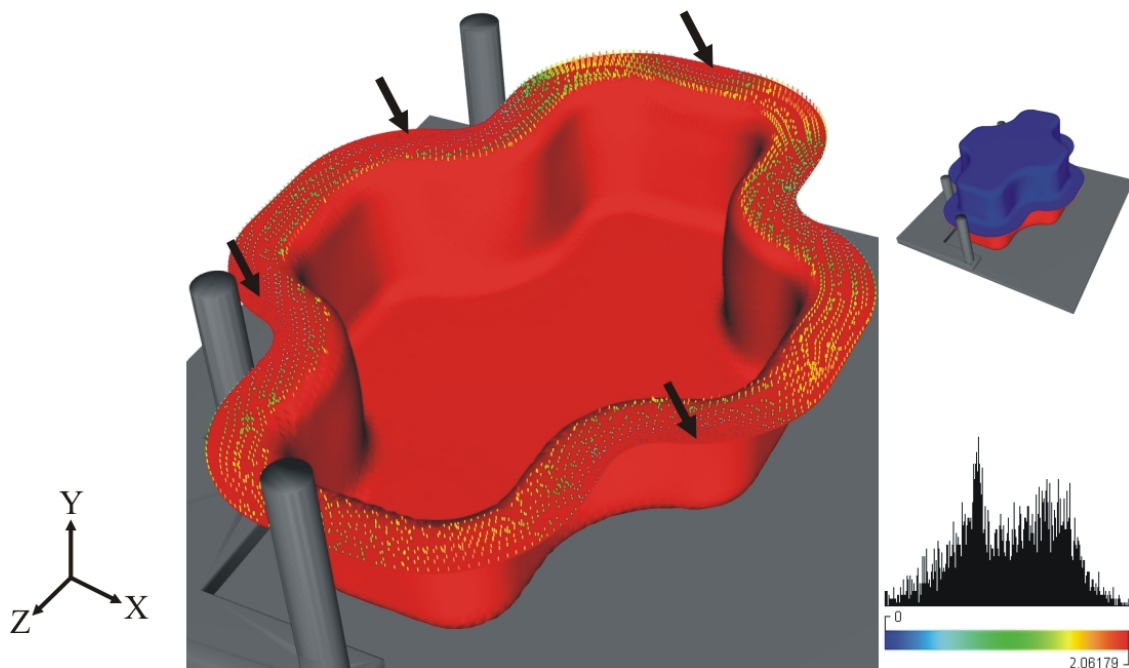


Bild 47: Abstandsmessung im Flanschbereich der positionierten Kreuzziehteile

Zu erkennen ist, dass die beiden Tiefziehteile sich nur an den Flanschrändern berühren. Die Auflagebereiche sind in Bild 47 durch Pfeile markiert. Das obere Bauteil (blau) ist ausgeblendet. Das Histogramm zeigt weiterhin, dass der Auflagebereich relativ klein ist: Nur wenige Zweipunktmessungen zwischen den Näpfen ergeben Werte unter 0,1 mm. Weiterhin sind zahlreiche gelbe und rote Messlinien sichtbar, d.h. an vielen Stellen treten im Flanschbereich Spaltabstände von 1,5 mm bis 2,06 mm auf.

Über eine Hilfsebene lässt sich außerdem die Gesamthöhe der Baugruppe visualisieren. Die Ebene wurde so platziert, dass sie auf Höhe der Oberseite der Nominalbaugruppe liegt. Gemessen wurde von der Bodenfläche des Napfes zur Hilfsebene.

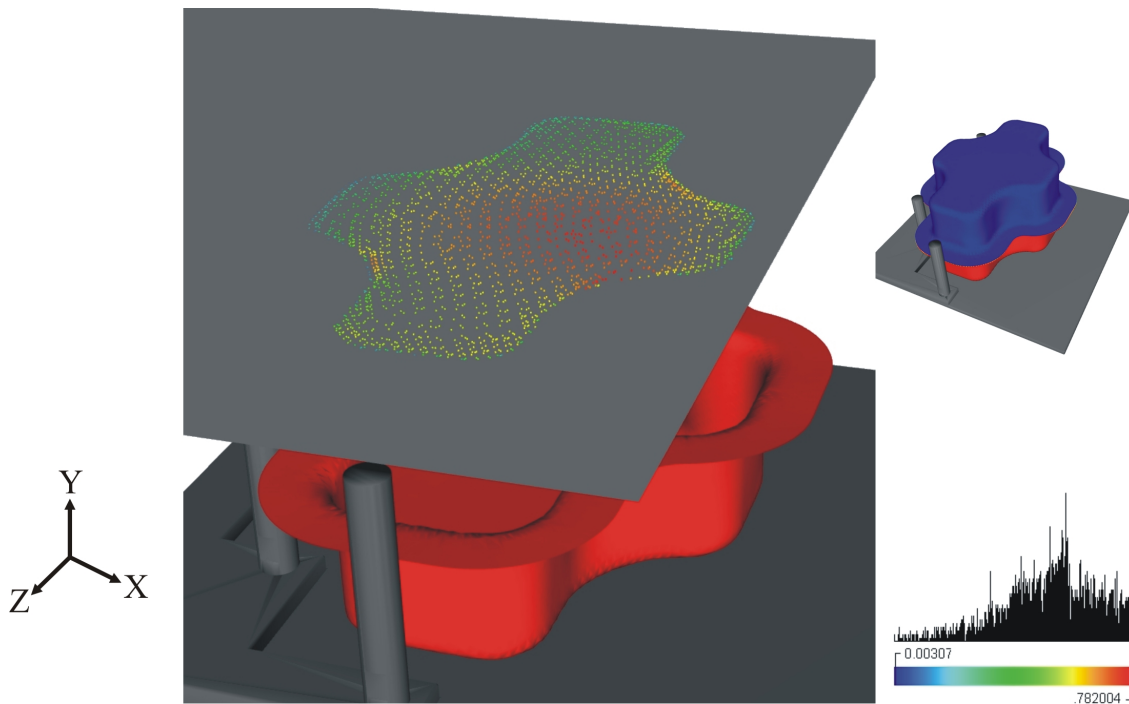


Bild 48: Höhenmessung der Napfbaugruppe

Die eingefügte waagrechte Hilfsebene ist in Bild 48 hellgrau dargestellt. Durch die Messung wird sichtbar, an welchen Stellen die positionierte Baugruppe ihre maximale Höhe besitzt. In Bild 48 ist das Gebiet durch rote Messstriche erkennbar. Dadurch wird ersichtlich, an welchen Stellen die nichtideale Baugruppe voraussichtlich ihre niedrigsten bzw. höchsten Bereiche besitzt. Zeigen die meisten analysierten Baugruppen dieses Merkmal auf, kann das Wissen frühzeitig, beispielsweise bei der Konstruktion von Dichtungen, genutzt werden. Neben den gezeigten Einsatzfeldern lässt sich die Visualisierung der Abstandsmessungen zur Analyse von Spaltverläufen, beispielsweise im Karosseriebau, einsetzen, siehe [93].

Zur Analyse einzelner nichtidealer Baugruppenvarianten wurde neben dem soeben beschriebenen Visualisierungsverfahren ein System zur haptischen Analyse entwickelt.

6.2 Haptische Analyse von nichtidealen Baugruppen

Unter den Begriff *Haptik* versteht man das aktive Erfühlen von Objekten. Unter dem Begriff „haptischer Wahrnehmungssinn“ wird bei Verwendung des Begriffs als Bezeichnung für ein User Interface „die Summe von taktilen Sinn (Rezeptoren der Haut) und kinästhetischem Sinn (Rezeptoren der Muskeln und Gelenke)“ verstanden, siehe [13]. Das Auflösungsvermögen der haptischen Wahrnehmung beim Menschen liegt bei 0,6 mm (Fingerkuppen) bis 0,1 – 0,2 mm (Handballen). Aufgrund dieser hohen Genauigkeit werden die Handballen oft von Designern zur Flächenbeurteilung verwendet, siehe [13].

Als Force-Feedback-Geräte bezeichnet man Systeme, die aus Sensoren und Aktoren bestehen. Die Sensoren registrieren Eingaben des Benutzers (z. B. Bewegungen der Hand) und geben sie an den verbundenen Rechner weiter. Die Aktoren geben daraufhin durch Computersimulation berechnete Kräfte an Benutzer weiter. Anhand der verwendeten Aktorik unterscheidet man aktive und passive Force-Feedback-Systeme: Aktive Systeme sind in der Lage, Kräfte direkt (z. B. durch Elektromotoren) zu erzeugen. Passive Systeme können lediglich Widerstände gegen die Bewegungen des Benutzers generieren.

Die folgenden Unterkapitel beschreiben ein entwickeltes Force-Feedback-System zur haptischen Toleranzanalyse. Nach einem Überblick zum Stand der Technik wird auf die grundsätzliche Architektur von haptischen Analysesystemen eingegangen. Im Anschluss wird das implementierte Haptiksystem vorgestellt und ermittelte Ergebnisse zur Genauigkeit bei Spalt- und Versatzmessungen zusammengefasst. Unterstützt wurden die Entwicklungen im Bereich Haptik durch studentische Arbeiten von Thomas Baumgärtner ([10]), Ludwig Otter ([91]) und Sven Albert ([3]). Auf dem Gebiet Physiksimulation bestand eine Kooperation mit dem wissenschaftlichen Mitarbeiter Klaus Iglberger (Lehrstuhl für Systemsimulation, FAU Erlangen-Nürnberg). Ein Zwischenstand der Arbeiten wurde in [136] veröffentlicht. Die erzielten Ergebnisse werden abschließend diskutiert und zusätzlich Weiterentwicklungsmöglichkeiten aufgezeigt.

6.2.1 Stand der Technik

Force-Feedback-Geräte wurden zunächst für die Anwendung in der Nuklear- und Tiefsee-Industrie entwickelt, siehe [121]. Durch sinkende Anschaffungskosten sind sie

mittlerweile auch in anderen Bereichen (z. B. Medizintechnik und Luftfahrt) anzutreffen. Seit einigen Jahren werden auch Systeme eingesetzt, die Konstrukteure bei der Produktentwicklung unterstützen. Hierbei wird hauptsächlich die Wartbarkeit der Produkte in virtuellen Umgebungen bei Montage- und Demontagesimulationen untersucht. Durch den Einsatz von virtuellen Prototypen sollen Fehler frühzeitig erkannt und reale Prototypen eingespart werden.

In [29] wird die Plattform SAMIRA beschrieben. Die Plattform verwendet ein Haptikgerät mit sechs Freiheitsgraden (Degrees of Freedom, DOFs) der Aktorik und Sensorik zur Simulation von Wartungsaufgaben. Damit das System von Konstrukteuren in der Praxis gewinnbringend einzusetzen ist und genutzt wird, müssen laut [29] folgende Bedingungen erfüllt sein: Die VR-Daten müssen durch eine konsistente, fest definierte Vorgehensweise aus den Produktdaten erstellt werden, Simulationsergebnisse schnell und auf einfache Weise generierbar sein und die Ergebnisse eine bekannte, hohe Genauigkeit besitzen. Nur wenn diese Bedingungen vollständig erfüllt sind, wird die Anwendung von Produktentwicklern eingesetzt. Andernfalls sind die Simulationsergebnisse möglicherweise veraltet oder nicht verlässlich.

In [74]) wird die Anwendung eines 3-DOF-Haptikgeräts zur Untersuchung von Ein- und Ausbauszenarien beschrieben. Um die fehlenden drei Freiheitsgrade der Aktorik zur vollständigen Objektmanipulation auszugleichen, wurden zwei unterschiedliche Benutzungsmodi implementiert: Im Translationsmodus kann das montierte Bauteil nur verschoben, im Rotationsmodus nur rotiert werden. Auf diese Weise können mit dem 3-DOF-Gerät Wartungssimulationen durchgeführt werden. Allerdings ist die Handhabung weniger intuitiv als bei 6-DOF-Geräten.

Zur Simulation der Festkörperphysik in Haptikanwendungen existieren zahlreiche Bibliotheken. Einen Überblick gibt [94]: Am weitesten verbreitet sind die Open-Source-Systeme ODE (siehe [108]) und Bullet (AMD). Erfolgreiche kommerzielle Systeme sind erhältlich von Havok (Havok Engine), Nvidia (PhysX), Boeing (VPS/PBM), CMLabs (Vortex) und Haption (IPSI).

6.2.2 Zielsetzungen

Zielsetzung des haptischen Analysesystems ist es, den Produktentwickler möglichst frühzeitig bei der Untersuchung von Produktdaten zu unterstützen. Hier lassen sich zwei Anwendungsklassen der haptischen Toleranzanalyse identifizieren:

- Die virtuelle Abtastung von montierten, nichtidealen Baugruppen zur Gewinnung von Flächeneigenschaften sowie Spalt- und Versatzmaßen und deren Verläufen. Hierbei werden Baugruppen hinsichtlich funktionaler und ästhetischer Eigenschaften durch das Bewegen eines Tastkörpers über die Geometrieoberflächen überprüft.
- Die Durchführung von Montageanalysen für toleranzbehaftete Bauteile. Vergleichbar ist dieser Einsatz mit einer „interaktiven“ Pfadplanung im Unterschied zu einer automatischen Pfadplanung wie in Kapitel 5 beschrieben.

Als Eingangsdaten für die haptische Analyse lassen sich hoch aufgelöste Dreiecksnetze verwenden, die bereits durch Relative Positionierung (vgl. Kapitel 4) kollisionsfrei zueinander angeordnet wurden. Die haptische Simulation muss so ablaufen, dass der Anwender den Eindruck hat, mit einer realistischen Umgebung zu interagieren. Die zu erfüllenden Eigenschaften des Systems lassen sich in die Bereiche Physiksimulation, Force Feedback und Visualisierung untergliedern.

Die bewegten Objekte müssen ein realistisches physikalisches Verhalten zeigen: Effekte wie Trägheit, Reibung und Gravitation sollten abbildbar sein. Ebenso wie in Kapitel 4 und 5 dürfen bewegte Objekte nicht ineinander eindringen, Kollisionen müssen realistisch gehandhabt werden. Die Antwortzeiten der Physiksimulation müssen sehr gering sein, damit der Eindruck von Interaktivität entsteht.

Die erzeugte Krafrückkopplung muss folgende Eigenschaften aufweisen: Sowohl die über Sensoren aufgezeichneten Benutzereingaben und die von der Physiksimulation berechneten Kräfte müssen über die Aktoren der Haptik (z. B. Servomotoren) ausreichend schnell verarbeitet und wiedergegeben werden können. Erst bei Antwortzeiten von 1 bis 2 ms (1000 bis 500 Hz) wird Force Feedback als glaubhaft wahrgenommen, siehe [94]. Zum Vergleich: Interaktive Grafik wird bereits bei 33 ms Abstand (bei 30 Bildern pro Sekunde) als flüssig empfunden.

Der Benutzer benötigt außerdem eine graphische Repräsentation der Szene, um seine Eingaben koordinieren zu können. Um im Analysesystem sowohl die Szene, als auch Abstände zwischen Tastkörper bzw. montiertem Bauteil und der Umgebung dreidimensional einschätzen zu können, ist die Visualisierung in einer stereographischen VR-Umgebung (Powerwall oder Cave) sinnvoll.

Insbesondere für die haptische Toleranzanalyse muss das Gesamtsystem weitere, spezielle Anforderungen erfüllen: Die Ergebnisse der Physiksimulation müssen eine

hohe Genauigkeit aufweisen. Anders als bei Computerspielen ist es nicht ausreichend, die simulierten geometrischen Objekte durch einfache Ersatzgeometrie anzunähern. Des Weiteren muss das physikalische Haptikgerät sehr geringes Spiel aufweisen und darf sich bei der Benutzerinteraktion nicht elastisch deformieren. Ansonsten ist es für den Anwender nicht möglich, zwischen simulierten Bauteilfehlern und Systemfehlern (Spiel, Deformation, Berechnungsungenauigkeit) zu unterscheiden.

Bild 49 fasst die Ein- und Ausgabeparameter, Randbedingungen und Störgrößen des Haptiksystems als Black-Box-Model zusammen.

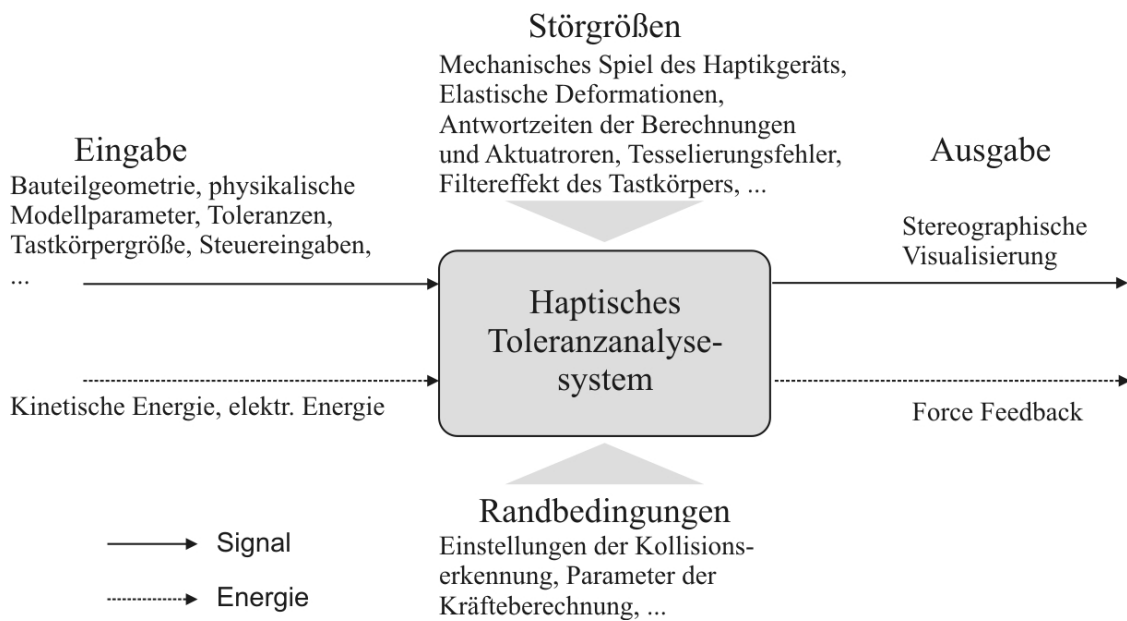


Bild 49: Black-Box Model des haptischen Toleranzanalyse-systems [136]

6.2.3 Aufbau des entwickelten haptischen Analysesystems

Basierend auf den beschriebenen Anforderungen wurde ein haptisches Toleranzanalyse-system aufgebaut. Es wurden folgende Systemkomponenten verwendet: Die „Haption Virtuose 3D-1525“ ist ein aktives Haptikgerät, d.h. es ist in der Lage, Kräfte zu erzeugen. Das Gerät besitzt Sensoren für sechs Freiheitsgrade (Handbewegungen in x-, y- und z-Richtung, Handrotation um drei Achsen). Die Aktoren besitzen drei Freiheitsgrade, d.h. translatorische Kräfte können durch Servomotoren erzeugt werden; Drehmomente können nicht generiert werden. Rotationen der Hand werden von den Sensoren (Potentiometern) ermittelt. Zur Benutzung wird der Griff des Geräts mit der Hand umfasst, Veränderungen der Handposition werden über optische Inkremental-Encoder detektiert. Zwei Druckknöpfe dienen zum Greifen und Neupositionieren virtueller Objekte.

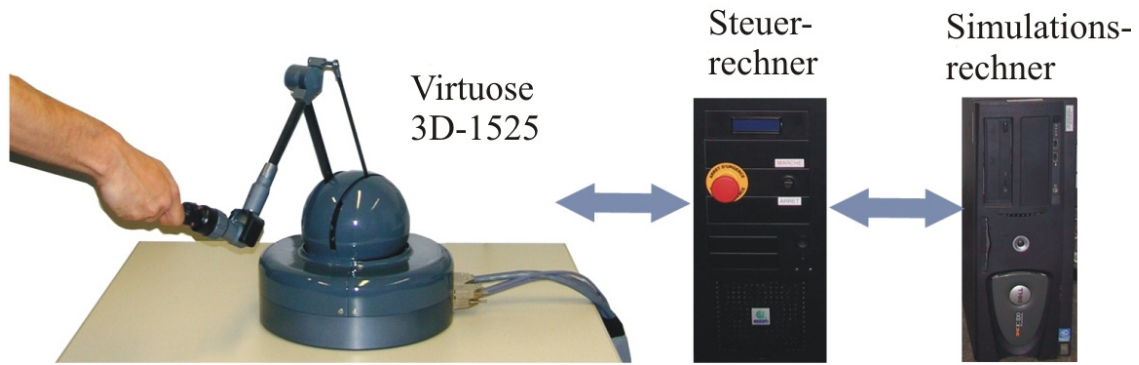


Bild 50: Kopplung der Hardwarekomponenten

Das Zusammenspiel der Systemkomponenten ist in Bild 50 abgebildet. Der Steuerrechner liest die Sensordaten des Haptikgeräts aus und sendet Steuersignale an die Servomotoren des Geräts. Auf einem Simulationsrechner finden die physikalischen Berechnungen statt, außerdem läuft auf diesem das Visualisierungs- und Analyseprogramm ab. Die frei verfügbare Open Dynamics Engine (ODE, siehe [108]) wurde verwendet, um die Starrkörperdynamik zu berechnen. In ODE können Körper entweder als Primitive (Kugeln, Zylinder, Box ...) oder komplexe Objekte (Dreiecksnetze) abgebildet werden. ODE führt die Kollisionserkennung und –behandlung durch und bestimmt die Objektbewegungen. Die Zielpositionen werden an den Steuerrechner weitergegeben, der wiederum die Servomotoren ansteuert. Die Open Dynamics Engine wurde dahingehend erweitert, dass der Import von Dreiecksnetzen im VRML-Format (siehe Kapitel 1.4.2) möglich ist.

Als Alternative zu ODE wurde die von Klaus Iglberger entwickelte PhysicsEngine PE (siehe [52], [53] und [54]) integriert. Diese erlaubt ebenfalls die Abbildung von Grundkörpern; Dreiecksnetze wurden zum aktuellen Zeitpunkt in PE noch nicht final implementiert. Die zu verwendende Engine kann als Startparameter der Haptiksimulation gewählt werden.

Zur Visualisierung der Szene wurde das OpenGL Utility Toolkit (GLUT) verwendet. Um dreidimensionale Bilder zu erzeugen, wird das Bild zwei Mal aus leicht unterschiedlichen Kamerapositionen gerendert (jeweils ein Mal für das linke und rechte Auge). Die Bilder werden von zwei Beamern mit Polarisationsfiltern auf eine Rückwand projiziert. Durch Betrachtung der Szene mit einer Brille, die ebenfalls Polarisationsfilter besitzt, erhält der Benutzer einen dreidimensionalen Eindruck der Szene. Dies erleichtert die Manipulation von Objekten und das Abschätzen von Abständen zwischen virtuellen Objekten.

6.2.4 Untersuchung der Genauigkeit des entwickelten Systems

Die haptische Analyse wird durchgeführt, indem die untersuchte Geometrie mit einem virtuellen Messkopf abgetastet wird. Der Benutzer steuert durch das Bewegen des Handgriffs der Haptik den virtuellen Messkopf. Auf diese Weise können Objekte (z. B. nichtideale Baugruppen) vom Anwender „berührt“ werden. Als Tastkörper wurde zunächst eine ideale Kugel gewählt. Da der Tastkörper bei manchen Untersuchungen im Vergleich der Szene sehr klein ist, lässt sich dessen Position teilweise schwer erkennen. Daher besteht die Möglichkeit, einen Kegel anzuzeigen, der ausschließlich graphisch repräsentiert ist und nicht in der Kollisionserkennung behandelt wird (siehe Bild 51, rechts).

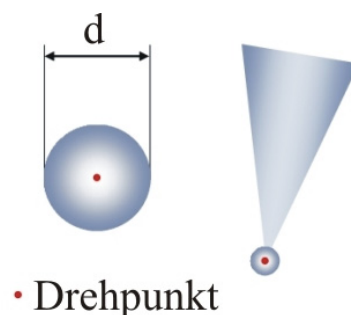


Bild 51: Tastkugel und Tastkörper mit Zeiger

Als Drehpunkt des Tastkörpers wurde der Mittelpunkt der Kugel gewählt. Auf diese Weise entstehen keine Probleme aus dem Umstand, dass die verwendete Haptik auf drei Freiheitsgrade der Aktorik beschränkt ist und keine Drehmomente erzeugen kann. Die Tastkugel kann während der Laufzeit verkleinert und vergrößert werden. So ist es möglich die Größe der Tastkugel auf die jeweilige Problemstellung (Abtasten eines Spaltverlaufs, Bewerten von Versatz zwischen Objekten etc.) anzupassen. Während der Simulation wird die Größe der Tastkugel in mm angezeigt.

Analyse von Spaltmaßen

Die haptische Analyse von Spaltmaßen ist eine der Hauptanwendungen des entwickelten Systems. Variierende Maße entlang von aneinandergrenzenden Teilen können sowohl hinsichtlich ihrer funktionalen als auch ästhetischen Eigenschaften untersucht werden. Quantitative Abstandsmessungen allein sind hierbei weder ausreichend, noch geeignet, um einen Eindruck von der Beschaffenheit des Spaltes zu bekommen.

Um die erreichbare Genauigkeit des implementierten Systems zu untersuchen wurde eine Szene, die, wie in Bild 52 dargestellt aus zwei nebeneinander befindlichen Primitiven „Box“ und einem Tastkörper „Kugel“ besteht, aufgebaut. Der Spalt zwischen den beiden Boxen wurde variiert, um die Szenarien „Vergrößerter Spalt“ und „Verkleinerter Spalt“ zu untersuchen. Der Durchmesser des Tastkörpers war für diese Untersuchung konstant.

Die virtuelle Geometrie der Szene kann beliebig skaliert werden, daher sind die angegebenen Messwerte der Experimente einheitenlos. Über den Parameter `virtSpeedFactor` kann das Verhältnis von realer Bewegung des Haptikarms und virtueller Bewegung des Tastkörpers eingestellt werden. In den Experimenten wurde dieser Parameter auf 1.5 gesetzt, eine reale Bewegung von 10 cm entspricht somit in der virtuellen Szene 15 cm.

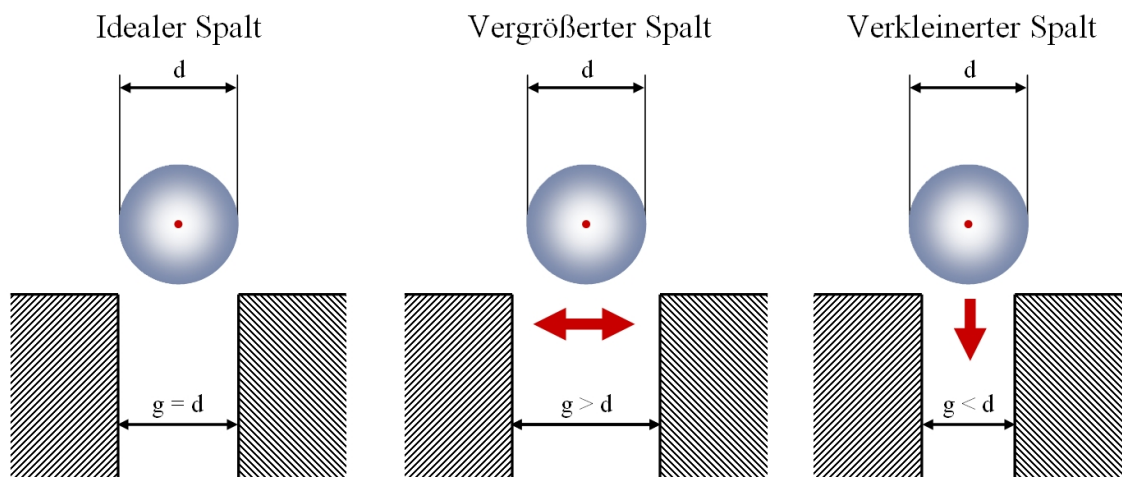


Bild 52: Analyse von Spaltmaßen

Bei der Untersuchung des verkleinerten Spalts ($g < d$) wurde überprüft, bis zu welchem Verhältnis von Spaltbreite g zu Tastkugeldurchmesser d sich die Tastkugel zwischen die Boxen bewegen lässt. Tabelle 15 zeigt die gemessenen Ergebnisse, vgl. [136]. Bei allen an der Haptik durchgeführten Untersuchungen ist zu beachten, dass die protokollierten Werte subjektiv sind. Zum einen hängen die Ergebnisse von der Benutzung des Systems ab, beispielsweise der Geschwindigkeit der Tastkörperführung und den vom Anwender aufbrachten Kräften. Zum anderen hängen die Ergebnisse von der Wahrnehmung des Anwenders ab, d.h. ab welcher Schwelle dieser die von den Servomotoren erzeugten Kräfte wahrnimmt. Nach einer gewissen Einarbeitungszeit werden jedoch durch verschiedene Benutzer ähnliche Werte erzielt, daher sind die erzielten Messungen trotz ihrer Subjektivität aussagekräftig.

Tabelle 15: Untersuchung, bis zu welcher Spaltbreite g sich eine Tastkugel mit Durchmesser d zwischen zwei ideale Boxen bewegen lässt

	ODE		PE	
Ø Tastkugel d	Spaltbreite g	Abweichung $(d-g)/d$	Spaltbreite g	Abweichung $(d-g)/d$
0,1	0,099964	0,036%	0,0999916	0,0084%
0,05	0,049965	0,07%	0,0499922	0,0156%
0,01	0,00976	0,24%	0,0099924	0,076%

Mit der Physikengine PE wurden wie in Tabelle 15 zu erkennen, höhere Genauigkeiten erreicht. Bei Anwendung der ODE lässt sich die Tastkugel durch Kraftaufwendung trotz der angegebenen Größenverhältnisse zwischen die Boxen bewegen, bei PE ist dies nicht möglich. Je kleiner die verwendete Tastkugel, desto ungenauer wird die Simulation des Kugelverhaltens in der Spaltanalyse.

Um festzustellen, ob der Tastkörper kleiner als die Spaltbreite ist ($g > d$), wurde der Abstand der Boxen so lange verkleinert, bis das Spiel des Tastkörpers durch den Anwender nicht mehr wahrnehmbar ist. Aufgrund des realen Spiels des Haptikarms ist dieser Messwert nicht in gleichem Maße wie in beim unterdimensionierten Spalt objektiv festzustellen.

Tabelle 16: Erkennbarkeit ein zu breiten Spaltmaßes g mit einem Tastkugeldurchmesser d

	ODE		PE	
Ø Tastkugel d	Spaltbreite g	Abweichung $(d-g)/d$	Spaltbreite g	Abweichung $(d-g)/d$
0,1	0,1017	1,7%	0,1014	1,4%
0,05	0,052	4,0%	0,0515	3,0%
0,01	0,0126	26,0%	0,0118	18,0%

Auch bei diesem Experiment zeigt die PE ein besseres Verhalten als ODE. Bei sehr kleinen Spalt- bzw. Kugelmaßen ist die prozentuale Abweichung bei beiden Engines sehr hoch (18% bzw. 26%).

Versatzanalyse

Unter dem Begriff Versatz versteht man die vertikale Verschiebung zwischen zwei Objekten. Zur Genauigkeitsuntersuchung wurde eine virtuelle Szene aufgebaut, in der zwei ideale Boxen direkt aneinander grenzen. Der Versatz wurde so lange variiert, bis er durch die Tastkugel nicht mehr wahrgenommen werden konnte. Unterschieden werden muss hierbei die Bewegungsrichtung der Tastkugel (siehe Bild 53): Wird diese auf die Stufe zu bewegt, so spürt der Anwender das Anstoßen gegen die Stufe. Bei einer Bewegung von A nach B nimmt der Anwender die Abwärtsbewegung der Kugel wahr.

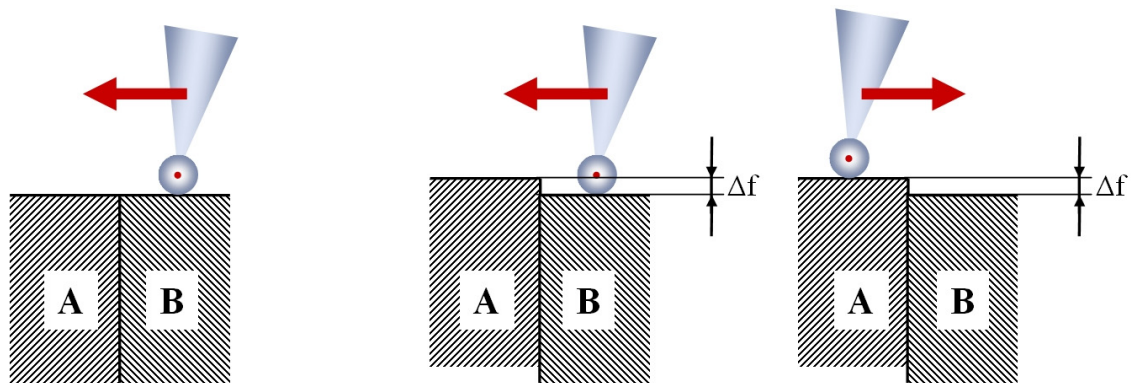


Bild 53: Analysefälle bei Versatzuntersuchungen

Bei Null-Versatz (Bild 53, links) sollte der Anwender den Übergang zwischen den Boxen A und B nicht wahrnehmen können. ODE gelingt dies, bei PE ist immer ein leichter Widerstand zu spüren. Dies lässt sich mit der Normalenberechnung von PE an der Kante der Boxen erklären. Trifft die Kugel an der Kante von A auf, steht der Normalenvektor leicht entgegen der Bewegungsrichtung der Kugel geneigt. Daher lässt sich die Versatzanalyse mit PE nicht durchführen. ODE erzielte bei der Versatzanalyse folgende Ergebnisse (vgl. [136]):

Tabelle 17: Wahrnehmbarer Versatz bei haptischer Analyse mit einem Tastkugeldurchmesser d

Ø Tastkugel d	Wahrnehmbares Δf bei Bewegung von B nach A	Wahrnehmbares Δf bei Bewegung von A nach B
0,1	0,0002	0,00009
0,01	0,0002	0,00003
0,001	0,0003	0,000004

Als Ergebnis ist festzuhalten, dass durch die Bewegung der Tastkugel entgegen der Stufe kleinere Versatzwerte ertastet werden können. Die Verwendung von großen

Tastkugeln führt zu einer Filterwirkung: Je kleiner die verwendete Tastkugel, desto geringere Versatzwerte können detektiert werden. Wichtig ist des Weiteren die Bewegungsgeschwindigkeit: Bei langsamer Abtastung können geringere Versatzwerte wahrgenommen werden, bei zu schneller Abtastung wird die Stufe (nicht wahrnehmbar) übersprungen.

6.2.5 Kombinierte Spalt- und Versatzanalyse

Bei realen Baugruppen tritt meist eine Kombination von Spalt- und Versatzabweichungen auf. Um derartige Geometrie zu untersuchen, wurde ein nichtidealer Zusammenbau, bestehend aus einer Motorhaube und einem Kotflügel, modelliert und ins haptische Analysesystem importiert. Visuell sind am Spaltverlauf zwischen den beiden Bauteilen keinerlei Abweichungen wahrzunehmen, siehe Bild 54. Durch extremes Heranzoomen wird die Abweichung erkannt, allerdings geht hierbei der Gesamtüberblick verloren und es kann keine Aussage mehr über den Spaltverlauf getroffen werden.

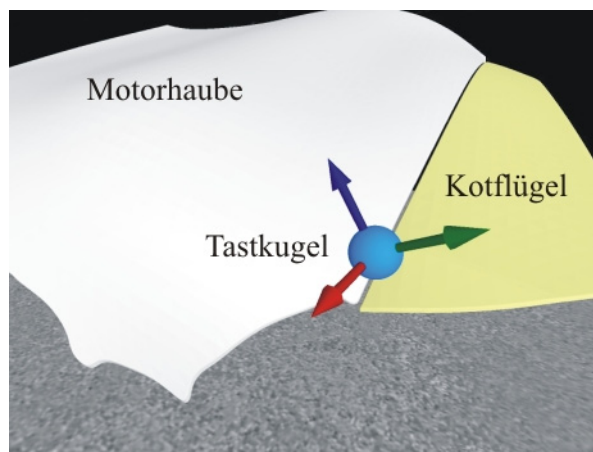


Bild 54: Haptische Analyse von Motorhaube und Kotflügel

Erst durch das Abtasten mit einer skalierbaren Tastkugel entlang des Spaltverlaufs können beispielsweise die dem Datensatz innewohnenden Abweichungen (z. B. kleine Erhöhungen) festgestellt werden. Trotz der geringen Größe der Geometriedefekte lassen sich diese deutlich wahrnehmen.

Es ist jedoch schwierig, die Stetigkeit des Übergangs zwischen Motorhaube und Kotflügel objektiv einzuschätzen. Spalt- und Versatzabweichungen treten gleichzeitig auf, daher sinkt die Tastkugel (je nach Durchmesser) bei der Stetigkeitsprüfung in den Zwischenspalt und bleibt folglich stets am gegenüberliegenden Bauteil hängen.

Des Weiteren hat sich ODE bei der Behandlung von Kollisionen zwischen Dreiecksnetzen und dem Primitiv „Kugel“ als nicht stabil erwiesen. Gelegentlich kommt es bei der Abtastung von Netzen zu nicht reproduzierbaren Abstürzen der Physikengine ODE.

Zur besseren Nutzbarkeit des Haptik-Systems wurden daher einige Erweiterungen implementiert: Um die Einschränkungen zu umgehen, die sich aus der fehlenden Moment-Erzeugung ergeben, wurde der Bedienmodus `rotationMode` entwickelt, bei dem das gegriffene Objekt nicht mehr translatorisch bewegt werden kann. Nach Definition eines Angriffspunkts können jedoch durch Translationen des Handgriffs Rotationen um den Objektmittelpunkt generiert werden. Stößt ein rotierendes Objekt an seine Umgebung, entstehen translatorische Kräfte, die über die Force Feedback an den Benutzer zurückgegeben werden können.

Um einen Tastkörper (z. B. einen Quader) eben auf eine Oberfläche aufzulegen, ist die Wiedergabe von Momenten durch das Haptik-System nötig. Ansonsten erzeugt der Benutzer durch unvermeidbare leichte Drehungen der Hand unbeabsichtigt Kräfte, die ein Abstoßen des Tastkörpers von der Oberfläche bewirken. Es ist haptisch nicht feststellbar, ob ein bewegter Quader eben auf einer abgetasteten Fläche aufliegt oder nur entlang einer Kante oder an einem Punkt aufsetzt.

Daher wurde ein sog. `dampingMode` implementiert: Dieser skaliert die von den Sensoren der Haptik detektierten rotatorischen Kräfte mit dem konstanten Faktor 0.05f. Dieser Wert wurde experimentell bestimmt. Die Skalierung bewirkt, dass sich das Tastobjekt bei Kollision mit einem fixen Körper an diesem ausrichtet. Über die Haptik lässt sich der Tastkörper dennoch steuern. Stößt der Tastkörper an einer virtuellen Kante an, so richtet er sich automatisch aus. Dieser Effekt ist nicht nur über das Force Feedback spürbar, sondern auch visuell zu erkennen.

Mit dem `dampingMode` lassen sich Versatzuntersuchungen auch für Bauteile durchführen, zwischen denen sich ein Spalt befindet. Prototypisch wurde dies für zwei nebeneinander platzierte Boxen durchgeführt. Der Benutzer ist in der Lage, den Tastkörper flach auf die erste Box aufzulegen und dadurch den Versatz der zweiten Box zu erkennen. Auch für komplexe, gekrümmte Geometrie war der Test des entwickelten Modus erfolgreich: Es wurde im CAD-System eine Baugruppe aus zwei aneinandergrenzenden Bauteilen erstellt, bei denen der Versatz im Spaltverlauf leicht zunimmt (siehe Bild 55). Der Anwender ist in der Lage, den nicht sichtbaren, langsam

ansteigenden Versatz zu ertasten. Durch die automatische Ausrichtung des Tastkörpers (siehe Markierung in Bild 55) lässt sich der Versatz auch graphisch erkennen.

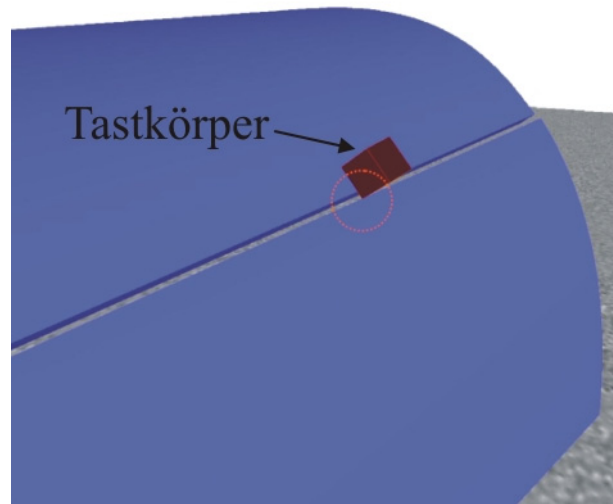


Bild 55: Versatzanalyse mit quaderförmigen Tastkörper (nach [3])

6.2.6 Montagesimulation nichtidealer Bauteile

Da der Anwender durch die Haptik in der Lage ist, intuitiv virtuelle Objekte in sechs Dimensionen zu manipulieren, bietet es sich an, Montageuntersuchungen mit nichtidealen Bauteilen interaktiv durchzuführen. Seit Version 0.11 unterstützt ODE die Behandlung von Trimesh-Trimesh Kollisionen, so dass in der Analyse komplexe Geometrie betrachtet werden kann.

Es wurden verschiedene Szenarien modelliert, um die Montage von Bauteilen zu simulieren. Einfache Aufgabenstellungen, z. B. die Montage eines Bolzens mit Radius als Montagehilfe in eine Bohrung mit Übermaß, ließen sich durchführen. Die zurückgegebenen Kräfte vermitteln ein realistisches Gefühl des Montagevorgangs.

Bei der Montage von vernetzten Bauteilen treten jedoch teilweise nicht plausible, sehr große Kollisionskräfte auf, so dass teilweise sogar die Sicherheitsabschaltung der Haptikmotoren aktiviert wird. Die Behandlung von Dreiecksnetzen in ODE ist noch nicht stabil genug um damit zuverlässig Montagesimulationen durchführen zu können. Je komplexer und feiner vernetzt die Geometrie ist und je geringer das Spiel zwischen den Fügepartnern, desto problematischer wird die Simulation mit ODE. Teilweise lässt sich das Problem durch ein Aufschaukeln von kleinen numerischen Fehlern erklären: Das montierte Bauteil wird durch die Kollisionsbehandlung zu weit in die entgegen gesetzte Richtung verschoben; dies verursacht eine noch weitere Verschiebung, siehe Bild 56.

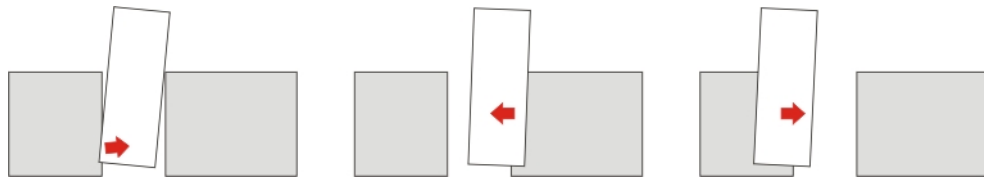


Bild 56: Aufschaukeln numerischer Fehler bei der Kollisionsbehandlung

Durch Modifikation relevanter ODE-Einstellungen (Integrationszeitschritt, Integrationsmethode, Kollisionsbehandlungsverfahren, Kontaktparameter) wurde das simulierte Verhalten der Objekte zwar verbessert. Eine Montagesimulation von komplexen Bauteilen mit geringen Abständen, wie in Kapitel 5 gezeigt, lässt sich durch die Kombination von ODE und dem vorhandenen Haptiksystem nicht durchführen. Bisher existiert keine Physiksimation, die in der Lage ist, die haptische Untersuchung von tolerierten Bauteilen mit der notwendigen Genauigkeit (Vernetzungsfehler deutlich unter 1 mm) und Antwortgeschwindigkeiten von 1 bis 2 Millisekunden durchzuführen. Dieses Ergebnis wurde von Jerome Perret (Haption) bestätigt [94].

6.2.7 Haptische Analyse: Zusammenfassung und Ausblick

In Kapitel 6.2 wurden Ansätze zur haptischen Analyse nichtidealer Baugruppen aufgezeigt. Die Hard- und Softwarekomponenten des hierzu entwickelten Systems und deren Zusammenspiel wurden eingehend beschrieben. Durch das Haptiksystem ist der Anwender in der Lage, Spalt und Versatz zwischen Bauteilen in Größenordnungen zu ertasten, die visuell nicht wahrnehmbar sind. Auch durch Heranzoomen können die gewünschten Aussagen (z. B. bezüglich Verlauf des Versatzes) nicht getroffen werden, da hierdurch die Übersicht verloren geht. Die haptische Analyse gibt dem Anwender ein neuartiges Werkzeug zur Hand, um nichtideale Baugruppen interaktiv zu untersuchen. Die erreichbare Genauigkeit der Spalt- und Versatzanalyse wurde in Experimenten für verschiedene Anordnungen von Grundkörpern gemessen. Es wurden Erweiterungen vorgestellt, die das fehlende rotatorische Force Feedback ausgleichen und damit das Verwenden von weiteren Tastkörpern und Stetigkeitsanalysen ermöglichen.

Eine Montagesimulation von komplexen, als Dreiecksnetz modellierten Bauteilen mit einer hohen Genauigkeit ist mit dem aktuellen System noch nicht durchführbar. Hier ist zu erwarten, dass mit zukünftigen Rechnergenerationen interaktive Physiksimation mit der geforderten Genauigkeit und Antwortgeschwindigkeit machbar wird.

6.3 Statistische Analyse durch Volumenvisualisierung

Um die Ergebnisse aus Toleranzsimulationen (z. B. aus Relativer Positionierung oder Monte-Carlo-Simulation) beurteilen zu können, steht dem Benutzer nach dem Stand der Technik die statistische Auswertung der numerischen Schwankung von fest definierten Messungen zur Verfügung. Die Abweichungswerte der Messungen werden meist in Verteilungskurven zusammengefasst, zusätzlich werden statistische Kenngrößen (z. B. Mittelwert, Standardabweichung etc.) gezeigt.

Einzelne nichtideale Baugruppen lassen sich des Weiteren wie in Kapitel 6.1 und 6.2 gezeigt, graphisch oder haptisch untersuchen. Bei der statistischen Toleranzsimulation entstehen jedoch sehr viele (Größenordnung 10^3 bis 10^4) Varianten der Baugruppe. Eine sequenzielle geometrische Analyse aller Varianten ist somit praktisch nicht umsetzbar. Kommerzielle Toleranzsimulationen bieten die Option an, die Varianten (ohne Formabweichung) in schneller Folge nacheinander anzuzeigen. Da dies von Toleranzanalyseexperten als äußerst unbefriedigend bewertet wird (Quelle: [37]), wurde ein spezialisiertes Visualisierungsverfahren zur statistischen geometrischen Analyse entwickelt, das in den folgenden Unterkapiteln vorgestellt wird. Das entwickelte Verfahren basiert auf einer Volumendarstellung der Daten.

Erste Ansätze zur statistischen Analyse von Bauteilabweichungen durch Volumenvisualisierung wurden in [135] und [137] präsentiert. Weiterführende Arbeiten zur Volumenvisualisierung wurden unterstützt von Florian Penzkofer, Marco Winter und Michael Martinek. Die Ergebnisse sind in den gemeinsamen Veröffentlichungen [93] und [115] dokumentiert.

6.3.1 Idee

Meine grundsätzliche Idee besteht darin, einen einzigen Datensatz zu generieren, der die Geometrie aller Bauteilvarianten einer statistischen Toleranzsimulation beinhaltet. Der naheliegende, in [78] von Löff et al. implementierte Ansatz, alle nichtidealen Varianten gleichzeitig zu laden und anzuzeigen, besitzt nur begrenzte Aussagekraft. Für den Anwender sind nur die Varianten sichtbar, die die äußere Hülle bilden. So können nur Extremlagen der positionierten Bauteile erkannt werden, es ist jedoch nicht ersichtlich, ob ähnlich positionierte Varianten häufig auftreten oder nur statistische Ausreißer an den Toleranzgrenzen darstellen.

Um diesen Nachteil zu beheben wurde ein Verfahren entwickelt, das zu jedem diskreten Raumbereich der Analyse die Wahrscheinlichkeit speichert, mit der sich an dieser Stelle

im analysierten Datensatz Geometrie befindet. Dies wurde über sog. Volumendaten realisiert. Nach Zusammenfassung der Geometriedaten durch Summierte Voxelisierung (Kapitel 6.3.2) kann das Ergebnis anhand spezieller Visualisierungsverfahren (Kapitel 6.3.3) dargestellt werden.

6.3.2 Erzeugen eines statistischen Datensatzes: Summierte Voxelisierung

Nach einer Erläuterung des Begriffs „Volumendaten“ wird im folgenden Text die Vorgehensweise der Summierten Voxelisierung dargestellt.

Die im Kapitel 4 und 5 vorgestellten Verfahren der Relativen Positionierung und Pfadplanung verarbeiten Bauteile, die als Dreiecksnetze dargestellt sind. Durch diese wird ausschließlich die Oberfläche von Objekten beschrieben, über den inneren Aufbau liegen keine Daten vor. Die in den nachfolgenden Unterkapiteln verwendeten Verfahren nutzen hingegen Volumendaten. Darunter versteht man Datensätze, die Informationen über den Inhalt jedes Raumbereichs speichern. Die nachfolgend betrachteten Volumendaten sind Skalarwerte, die beispielsweise in einem uniformen Gitter angeordnet sind. Als Analogie zum Volumendatensatz kann ein zweidimensionales digitales Bild herangezogen werden: Das 2D-Bild ist unterteilt in gleichgroße rechteckige Bereiche, sogenannte Pixel. Jedes Pixel enthält Daten, beispielsweise einen Farbwert. Analog dazu ist ein Voxel die kleinste beschreibbare kubische Raumeinheit des Volumens. Jeder Voxel speichert wiederum einen Wert.

Die aus der Toleranzsimulation stammenden Geometriedaten werden in Volumendaten übersetzt, indem jede Baugruppe in die Voxel-Darstellung überführt wird. Erzeugt wird ein einziger Volumendatensatz für alle Geometrievarianten. Das entwickelte Umwandlungsverfahren, als Summierte Voxelisierung bezeichnet, läuft folgendermaßen ab (siehe Bild 57):

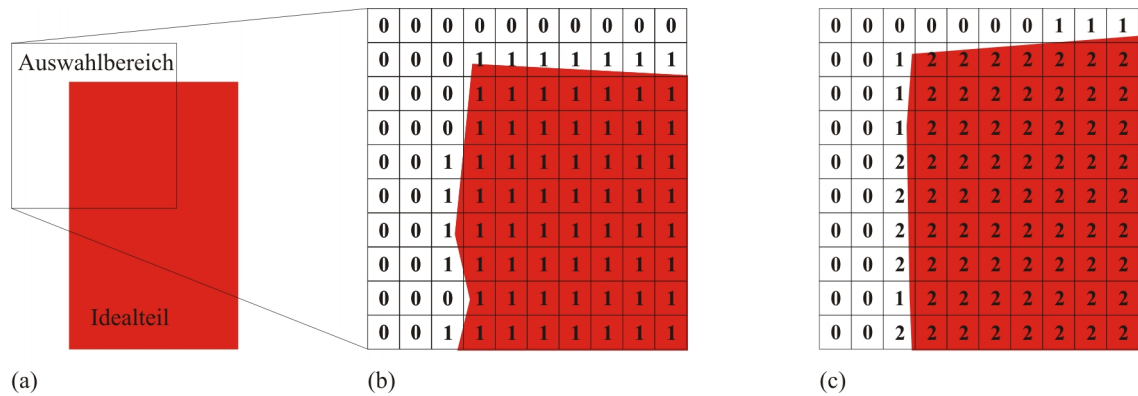


Bild 57: Ablauf der Summierten Voxelisierung am Beispiel von zwei nichtidealen Baugruppenvarianten

Der Benutzer wählt einen Bereich der untersuchten Baugruppe zur Analyse aus, in Bild 57(a) beispielsweise die linke obere Ecke. Für diesen Bereich wird ein Volumendatensatz angelegt. Zu Beginn wird der Speicherwert jedes Voxels des Datensatzes auf den Wert „0“ gesetzt. Im Anschluss wird die erste nichtideale Variante der analysierten Baugruppe geladen.

Bei jedem Voxel, der von dieser Variante geschnitten wird, wird der Zählerwert auf „1“ erhöht. Die restlichen Voxel bleiben unverändert auf „0“, siehe Bild 57(b). Ist die Voxelisierung der ersten Variante abgeschlossen, wird die nächste nichtideale Variante geladen. Der Zähler aller diesmal geschnittenen Voxel wird abermals erhöht, so dass danach im Summierten Volumendatensatz die Voxelwerte 0, 1 und 2 vorliegen, je nachdem wie oft der jeweilige Voxel geschnitten wurde vgl. Bild 57(c). Der Intersektionsprozess wird für alle nichtidealen Baugruppenvarianten durchgeführt. Nach der Voxelisierung von n Varianten enthält der Summierte Volumendatensatz Voxel mit den Trefferzahlen 0 (keine Variante schneidet) bis maximal n (alle Varianten schneiden). Wird die Trefferzahl durch n dividiert, lässt sich das Ergebnis als Aufenthaltswahrscheinlichkeit (relative Häufigkeit) der Geometrie im jeweiligen Raumbereich interpretieren.

Zur Durchführung der Voxelisierung muss der Benutzer eine Kantenlänge der Voxel vorgeben, die für die auftretenden Abweichungen der analysierten Aufgabenstellung fein genug ist. Wird die Kantenlänge zu groß gewählt, verteilen sich die Bauteilvarianten auf wenige Voxel und zeigen nur eine grobe geometrische Verteilung. Bei zu kleiner Kantenlänge werden Speicherbedarf und Rechenzeit so groß, dass Voxelisierung und Weiterverarbeitung problematisch werden. Für jeden Voxel muss die Trefferzahl im Speicher (z. B. als 16-Bit-Wert) abgelegt werden. Der Speicherbedarf des Volumendatensatzes verhält sich kubisch zur Kantenlänge, das heißt eine

Halbierung der Voxel-Kantenlänge verachtfacht den Speicherbedarf. Der Benutzer muss somit einen Kompromiss zwischen möglichst feiner Auflösung und vertretbarem Speicher- bzw. Verarbeitungsaufwand wählen. Da aufgrund dieser Problematik nicht beliebig große Bereiche (z. B. eine vollständige Autotür) mit hoher Auflösung voxelisiert werden können, wird der Analysebereich in der entwickelten Anwendung durch eine achsenparallele Auswahlbox, wie in Bild 58 gezeigt, durch den Anwender eingeschränkt.

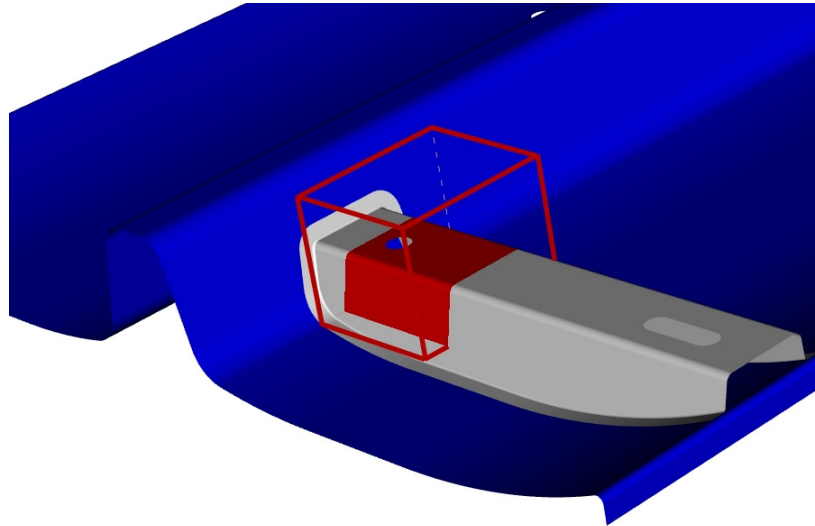


Bild 58: Festlegung des Voxelisierungsbereichs durch eine Auswahlbox

Bei der Voxelisierung existieren zwei alternative Vorgehensweisen. Es kann sinnvoll sein, nur die Oberfläche der toleranzbehafteten Geometrie zu betrachten. Hierzu wird nur die Bauteiloberfläche mit den Voxeln geschnitten. Andererseits kann das gesamte Innere der Geometrie ebenfalls in Voxel umgewandelt werden. Die Bauteile werden damit als „Solids“ behandelt. Umgesetzt wurden in der erstellten Anwendung beide Voxelisierungsarten für einen Regelgeometrietyp sowie für allgemeine Dreiecksnetze. Als Regelgeometrie wurde beispielhaft die Voxelisierung des Geometrietyps „Zylinder“ implementiert. Zylinder lassen sich beispielsweise durch ihre Achsenendpunkte und ihren Durchmesser beschreiben. Für jeden Voxel des Volumendatensatzes wird getestet, ob er vom Zylindermantel geschnitten wird (Oberflächenvoxelisierung) bzw. ob er innerhalb des Zylinders liegt (Solid-Voxelisierung). Details der angewendeten Verfahren zur Schnittberechnung zwischen Voxel und Zylinder sind in [92] beschrieben. Um die Geschwindigkeit der Voxelisierung zu erhöhen, wurde ein Verfahren implementiert, das die Abstandsberechnungen hierarchisch für Blöcke aus mehreren Voxeln durchführt. Durch die Blockteilung ließ sich die Rechenzeit stark verringern.

Für Dreiecksnetze als universelle Form der Geometriebeschreibung wurden ebenfalls Voxelisierungsverfahren entwickelt. Eine Oberflächenvoxelisierung von Dreiecksnetzen wurde dadurch realisiert, dass für jedes Dreieck die Axis-Aligned Bounding Box (AABB) erstellt wurde. Alle Voxel innerhalb der AABB lassen sich anhand des in [1] gezeigten Verfahrens effizient auf Intersektion testen. Damit der Trefferzähler jedes Voxels maximal ein Mal pro nichtidealer Variante erhöht wird, wird jeweils zusätzlich ein Bit als Dirty-Flag gespeichert. Der Zähler wird nur erhöht, wenn das Bit noch nicht gesetzt ist. Eine Solid-Voxelisierung von Dreiecksnetzen lässt sich CPU-basiert mit Punkt-Abfragen durchführen wie in [75] gezeigt. Höhere Voxelisierungsgeschwindigkeiten bietet allerdings die Verarbeitung durch die Grafikkarte anhand spezieller Fragment-Shader, wie z. B. in [30] beschrieben.

6.3.3 Ergebnisvisualisierung

Um einen Datensatz zu analysieren, der wie in Kapitel 6.3.2 beschrieben durch Summierte Voxelisierung erzeugt wurde, bieten sich Visualisierungsverfahren an. Aus der Computergraphik stehen dafür zahlreiche Techniken für Volumendaten zur Verfügung, die zu diesem Zweck genutzt werden können.

Nach der Summierten Voxelisierung und Normierung der Daten besitzt jeder Voxel einen Skalarwert im Bereich 0 bis 1, der graphisch beliebig interpretiert werden kann. Die Zuordnung von graphischen Eigenschaften geschieht über eine Transferfunktion. Der Skalarwert wird durch die Transferfunktion beispielsweise in Farbwert, Transparenzwert und weitere graphische Eigenschaften übersetzt. Jedem Voxelwert lässt sich beispielsweise ein Transparenzwert zuweisen, so dass niedrige Werte zu einem nahezu durchsichtigen (transparenten) Voxel führen und hohe Werte zu einem undurchsichtigen (opaken) Voxel. Bild 59 zeigt beispielhaft die Visualisierung eines Computertomographie-Datensatzes. Auf der rechten Seite von Bild 59 sind anhand der Transferfunktion alle Bereiche mit niedrigen Voxelwerten (entspricht hier einer niedrigen Röntgendichte) ausgeblendet. Visualisierungsbeispiele aus dem Maschinenbau folgen in Unterkapitel 6.3.5

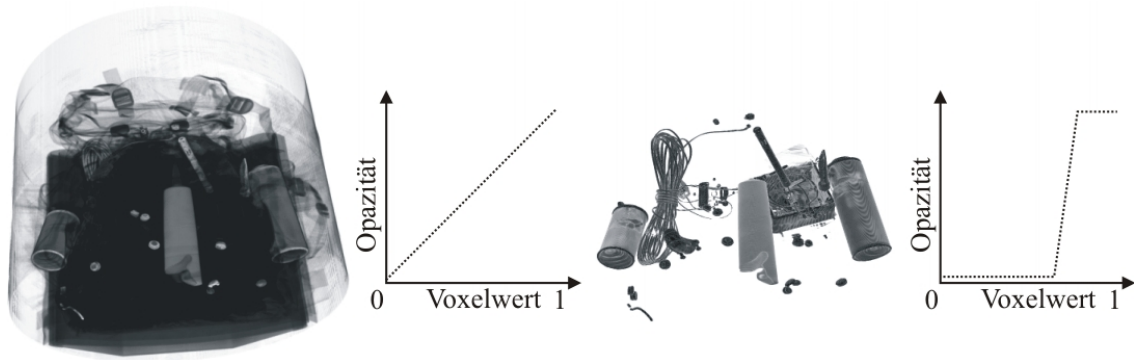


Bild 59: Visualisierung des CT-Datensatzes „Rucksack“ (Datenquelle: [62]) mit unterschiedlichen Transferfunktionen. Die Diagramme zeigen jeweils die Zuweisung der Opazität zum Voxelwert

Über die Definition der Transferfunktion erhält der Anwender somit die Möglichkeit, die Daten zu klassifizieren. Wertebereiche können durch entsprechende Farb- und Transparenzzuordnung sowohl hervorgehoben als auch ausgeblendet werden. Das transparente Ausblenden ist bei der Toleranzanalyse beispielsweise für Voxel ohne Treffer oder mit sehr geringen Trefferzahlen einsetzbar.

Je nach Anwendungsfall muss die Transferfunktion neu definiert werden. Allerdings besteht die Möglichkeit, erstellte Transferfunktionen z. B. als eindimensionale Textur abzulegen, die pro Pixel RGB- und Alphawert enthält. So können Transferfunktionen gespeichert und geladen werden.

Direkt darstellen lässt sich das Volumen beispielsweise, indem das Volumen als 3D-Textur in den Speicher der Grafikkarte geladen wird. Über eine hohe Anzahl an Schichten, die durch das Volumen gelegt und texturiert angezeigt werden, lässt sich der Datensatz mit interaktiven Bildwiederholraten untersuchen. Limitierender Faktor ist bei dieser Visualisierungsart der verfügbare Texturspeicher. In [101] und [129] sind Verfahren zur Volumenvisualisierung durch Grafikkarte detailliert beschrieben.

Indirekte Visualisierungsverfahren lassen sich zur Analyse des Summierten Volumendatensatzes ebenfalls einsetzen. Beispielsweise wird bei der Schichtdarstellung eine achsparallele oder frei ausgerichtete Ebene mit dem Volumen geschnitten, wobei ein 2D-Schnittbild entsteht. Marching Cubes (siehe [80]) hingegen extrahiert aus einem Volumendatensatz zusammenhängende Oberflächen mit dem gleichen Schwellwert.

6.3.4 Volumenbasierte statistische Kollisionserkennung

Anhand der bisher beschriebenen Summierten Voxelisierung kann die räumliche Schwankung einer einzelnen Baugruppe untersucht werden. Aufgrund funktionaler Anforderungen an die untersuchte Geometrie besteht oft zusätzlich Interesse daran, das

Verhalten von Baugruppen zueinander zu analysieren. Insbesondere mögliche Kollisionen, z. B. aufgrund von nichtidealer Positionierung und abweichungsbehafteter Einzelteilgeometrie, können mit der folgenden Erweiterung der statistischen Volumenvisualisierung graphisch dargestellt werden.

Verarbeitung unabhängiger Baugruppen

Sind die betrachteten Baugruppen A und B nicht voneinander abhängig, d.h. beeinflussen sie sich nicht gegenseitig bei ihrer Positionierung, wird folgender Ansatz der Kollisionsberechnung vorgeschlagen:

Da ein durch Summierte Voxelisierung erstellter Volumendatensatz die Trefferzahlen der Geometrie für jeden Voxel enthält, kann aus zwei entsprechenden Volumendatensätzen V_A und V_B , die einen Bereich der aneinandergrenzenden Baugruppen A und B repräsentieren, die statistische Kollisionswahrscheinlichkeit pro Voxel für den Datensatz berechnet werden. V_A und V_B müssen sich zur Kollisionsbestimmung überschneiden, wobei das implementierte Verfahren annimmt, dass V_A und V_B den gleichen Auswahlbereich (vgl. Bild 58) repräsentieren und die Kantenlänge der Voxelisierung identisch ist. Weiterhin müssen die Volumen aus einer Solid-Voxelisierung stammen.

Für das Ereignis K_x = Kollision von A und B in Voxel x beträgt die Wahrscheinlichkeit

$$P(K_x) = P(A_x) \cap P(B_x) = P(A_x) \cdot P(B_x)$$

wobei die Wahrscheinlichkeiten

$P(A_x)$ = Wahrscheinlichkeit, mit der A Voxel x schneidet

$P(B_x)$ = Wahrscheinlichkeit, mit der B Voxel x schneidet

implizit durch die Trefferzahlen von V_A und V_B gegeben sind.

Somit lässt sich die Kollisionsberechnung für jeden Voxel durch Multiplikation der Aufenthaltswahrscheinlichkeiten berechnen. Ein Vorteil dieser Kollisionsbestimmung besteht darin, dass die Kollisionswahrscheinlichkeit der Bauteile für jeden Voxel in einem einzigen Rechenschritt (durch die Überlagerung von zwei Summierten Volumendatensätzen) gewonnen wird. So können nach der Voxelisierung vergleichsweise schnell die statistischen Kollisionswahrscheinlichkeiten bestimmt und mittels Transferfunktion visualisiert werden.

Verarbeitung abhängiger Baugruppen

Bei statistischer Abhängigkeit von Baugruppe A und B ist eine paarweise Kollisionsprüfung nötig. Für jede zu testende Kombination von A und B müssen zunächst die Volumen V_A und V_B erstellt werden, die jeweils nur eine einzige geometrische Variante repräsentieren. Die Voxel dieser Volumen enthalten folglich jeweils entweder den Wert 0 oder 1. Die entstehenden Volumen V_A und V_B werden paarweise überlagert. Eine Kollision in Voxel x liegt vor, wenn sowohl V_A und V_B in Voxel x den Wert 1 besitzen. Die Überschneidung wird für jede mögliche Paarung von A mit B berechnet und in einem weiteren Volumendatensatz aufsummiert.

Einfluss des Diskretisierungsfehlers auf die Kollisionserkennung

Zu beachten ist, dass durch die in Kapitel 6.3.2 beschriebene Voxelisierung ein Diskretisierungsfehler entsteht. Voxelisierte Gegenstände nehmen dadurch stets mehr Raum ein als die Originalgeometrie. Zwei Bauteile, die sich nicht überschneiden, können gegebenenfalls als kollidierend erkannt werden, wenn sie einen Abstand besitzen, der kleiner ist als die Diagonale eines Voxels, siehe Bild 60. Mit der gewählten Vorgehensweise wird sichergestellt, dass auftretende Kollisionen auf jeden Fall erkannt werden. Alternativ ließe sich eine Voxelisierungsmethode implementieren, die Voxel nur als „belegt“ interpretiert, wenn sie vollständig in der Geometrie enthalten sind. Hierdurch würden jedoch manche Kollisionen nicht erkannt werden.

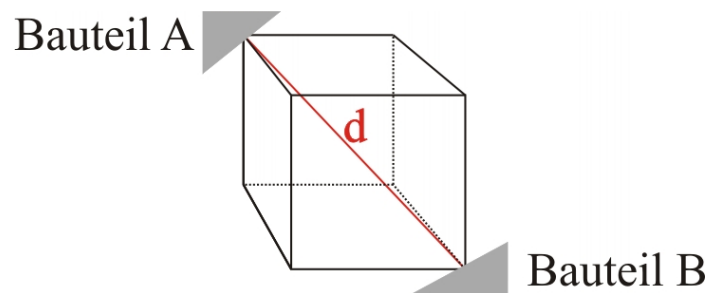


Bild 60: Bauteil A und B schneiden den gleichen Voxel, kollidieren jedoch nicht

Bei einer Voxel-Kantenlänge von a entspricht der Abstand d (aufgrund der Würfelform des Voxels) dem Wert, bei dem fälschlicherweise Kollisionen erkannt werden

$$d = \sqrt{3}a$$

Analog zur Summierten Voxelisierung muss der Anwender bei der volumenbasierten statistischen Kollisionserkennung die Kantenlänge a so klein wählen, dass trotz des Diskretisierungsfehlers eine sinnvolle Analyse der jeweiligen Baugruppe möglich ist.

Ergebnisdarstellung

Wurden die Kollisionswahrscheinlichkeiten für jeden Voxel bestimmt, so kann der resultierende Datensatz mit den in Kapitel 6.3.3 gezeigten Verfahren visualisiert werden, beispielsweise durch Zuweisung einer Transferfunktion. Kapitel 6.3.5 zeigt die Anwendung der Kollisionsanalyse auf Beispieldaten.

6.3.5 Experimentelle Ergebnisse

Anwendung und Ergebnisse der entwickelten Analysemethode werden in den folgenden Beispielen aufgezeigt. Dazu werden sowohl Dreiecksnetze als auch Regelgeometrie (Zylinder) als Eingangsdaten des volumenbasierten Verfahrens verwendet.

Statistische Volumenvisualisierung von Dreiecksnetzen

In Kapitel 4.5.5 wurde die Relative Positionierung von simulierten Kreuzzieh-Bauteilen beschrieben, die als detaillierte Dreiecksnetze modelliert sind. Durch die Kombination von 74 Bauteilnetzen wurden 5.476 nichtideale Baugruppen erstellt. Bild 45 auf Seite 98 zeigt die Analyse einer einzelnen Baugruppenvariante. Um die räumliche Schwankung aller simulierten Baugruppenvarianten zu untersuchen, wurde eine Summierte Voxelisierung des oberen Napfes erstellt. Zur Speicherung der Trefferzahlen wurden pro Voxel 16 Bit verwendet. Aufgrund des hohen Speicher- und Rechenbedarfs wurde eine Voxelisierung und Visualisierung eines Teilbereichs durchgeführt, der in Bild 61 dargestellt ist. Anhand des Ausschnitts können bereits zahlreiche Aussagen über das Schwankungsverhalten der Baugruppe getroffen werden. Mit zunehmender Rechenleistung wird es zukünftig möglich sein, auch große Baugruppen in einem hoch aufgelösten Volumen zu untersuchen.

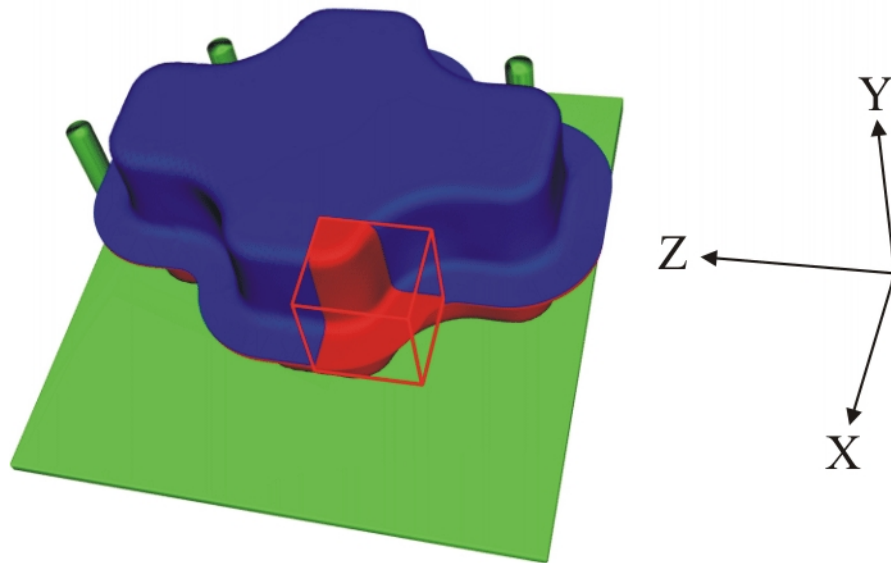


Bild 61: Festgelegter Voxelisierungsbereich am oberen Kreuzziehteil (roter Würfel)

Der resultierende Volumendatensatz besteht aus 512^3 Voxeln, die Kantenlänge eines Voxels beträgt 0,115 mm. Bild 62 zeigt die erstellte Volumenvisualisierung. Am linken Rand ist die Farbgebung der Transferfunktion dargestellt. Grün eingefärbte Voxel besitzen eine geringe Trefferzahl, rot gefärbte werden häufig durch die Geometrie des oberen Napfes geschnitten. Voxel, die von 100% der Nöpfe geschnitten wurden, sind zur besonderen Hervorhebung schwarz eingefärbt.

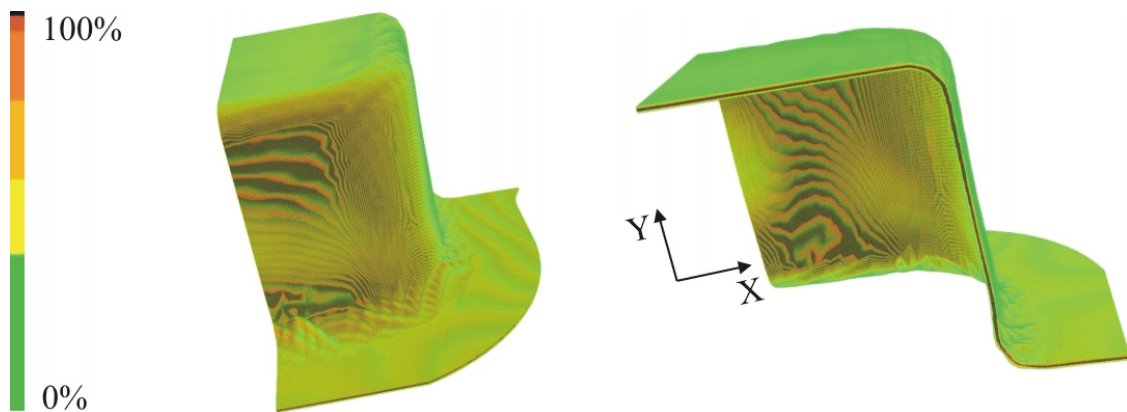


Bild 62: Volumenvisualisierung des gewählten Bereichs der 5476 nichtidealen Baugruppen

Durch die Visualisierung des Summierten Volumendatensatzes (siehe auch Bild 63 und Bild 64) lässt sich erkennen, dass die entstandenen Baugruppen hauptsächlich in y-Richtung variieren. Der in Bild 63 dargestellte Schnitt zeigt (durch kleine rote Pfeile angedeutet), dass die Seitenwände des Napfes in der x-y-Ebene nur geringfügig schwanken. Ein Großteil der Voxel ist im Wandbereich schwarz eingefärbt (siehe Schnitte in Bild 63 und Bild 64), besitzt also eine Trefferwahrscheinlichkeit von 100%. Geometrische Abweichungen zeigen sich vor allem im Flanschbereich und an der Decke des Kreuzziehtails (große Pfeile in Bild 63, rechts).

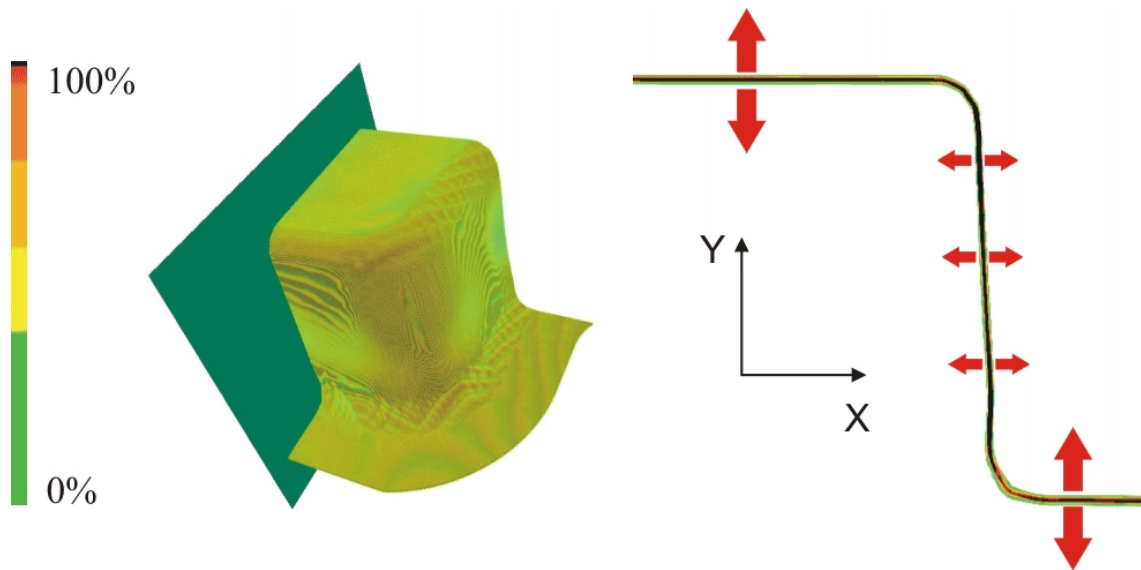


Bild 63: x-y-Schnitt des Volumens, Schnittebene in grün dargestellt

Die anliegende zweite Seitenwand des Kreuzziehteils zeigt in der Schnittdarstellung neben Schwankungen im Flansch- und Deckenbereich zusätzlich größere Schwankungen im Mittelbereich der Napfwand, siehe Bild 64. Dies ist fertigungstechnisch plausibel, da hier ein Bereich großer Ausdünnung liegt. Der Bereich wird maßgeblich durch den Anisotropiewert in Walzrichtung bestimmt, der für den gewählten Werkstoff mit 1,9 sehr hoch liegt. Dadurch entsteht die in Bild 64 sichtbare hohe Variation der Seitenwand.

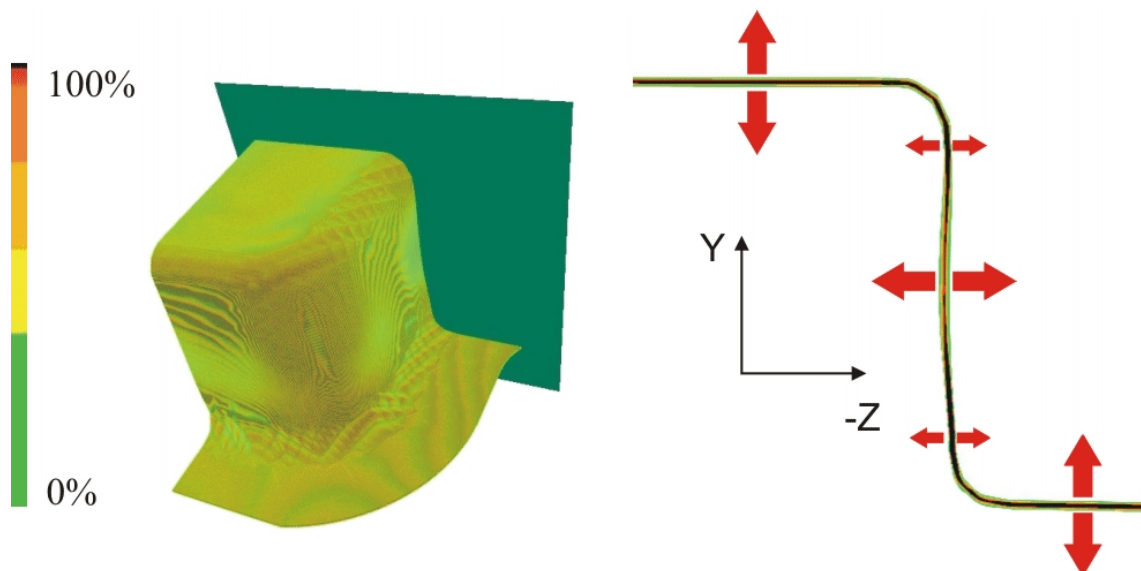


Bild 64: y-z-Schnitt des Volumens mit großen Schwankungen im Wandbereich

Die Volumenvisualisierung offenbart die räumliche Schwankung des gewählten Ausschnitts der Flanschbaugruppe. Zur Qualitätssicherung ist es somit sinnvoll, nach der realen Fertigung Messungen in den Bereichen der größten geometrischen Schwankung durchzuführen. Mit herkömmlichen Methoden der Punktmessung wäre es nicht möglich gewesen, diese Bereiche zuverlässig zu identifizieren.

Statistische Volumenvisualisierung von Regelgeometrie (Zylindern)

Von der Firma Bosch wurden freundlicherweise Simulationsdaten von Injektorspitzen und den umliegenden Injektorbohrungen zur Verfügung gestellt. Die Daten stammen aus dem kommerziellen Toleranzsimulationsprogramm eM-TolMate und wurden nach dem in Kapitel 2.2.2 beschriebenen Monte-Carlo-Verfahren generiert. Sie repräsentieren lediglich den untersten Ausschnitt einer umfassenden Toleranzuntersuchung des Injektoreinbaus in den Zylinderkopf. Von jedem der zylinderförmigen Bauteile liegen $n=10.000$ Varianten vor. Modelliert sind die Injektorspitzen bzw. Bohrungen im analysierten Datensatz als ideale Zylinder mit variierenden Achsenendpunkten und Durchmessern. Die Form und Lage der Injektorspitzen ist unabhängig von der umliegenden Bohrungsgeometrie. Die Lage des gesamten Injektors wird vor allem durch die Toleranzen des Führungszyinders und der Stufenbohrungen im Zylinderkopf bestimmt. Dem Anwendungsbeispiel kommt eine besondere Bedeutung zu, da Größenverhältnisse und Toleranzen in etwa den realen Verhältnissen der Fertigung entsprechen.

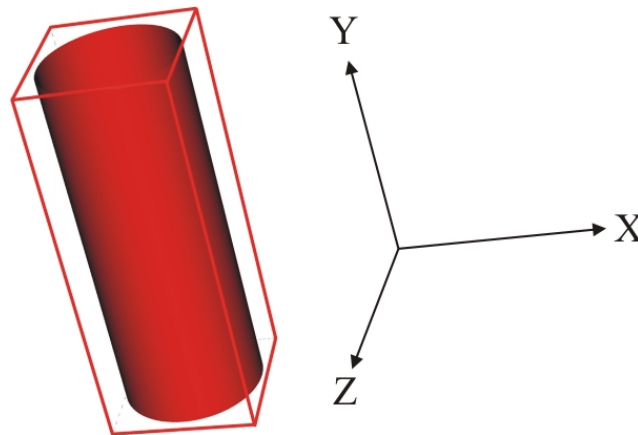


Bild 65: Voxelisierungsbereich und Koordinatensystem des Injektors

Zunächst wurde aus allen 10.000 Varianten der Injektorspitze durch Summierte Voxelisierung ein Volumendatensatz erstellt und hierbei die Oberfläche des Bauteils (die Zylindermantelfläche) zur Erstellung verwendet. Die Auflösung der Volumen betrug 333 Voxel in x- und z-Richtung, sowie 866 Voxel in y-Richtung. Aus Gründen der Geheimhaltung können die genauen Maße der Voxel nicht genannt werden. Bild 65 zeigt den Auswahlbereich der Voxelisierung und das verwendete Koordinatensystem.

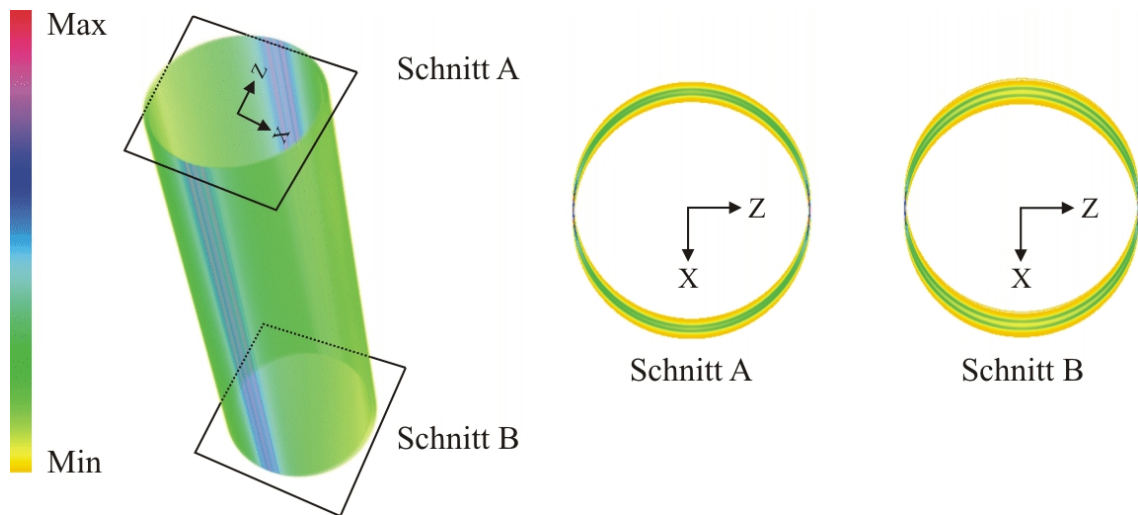


Bild 66: Volumendarstellung der Injektorhülle und Schnitte durch das Volumen

In Bild 66 ist die Volumendarstellung der Injektordaten abgebildet. Die zugewiesene Transferfunktion weist Voxeln mit hoher Trefferwahrscheinlichkeit die Farbe rot zu. Voxel mit niedrigen Trefferwahrscheinlichkeiten werden gelb eingefärbt.

Durch die Volumenvisualisierung wird ersichtlich, dass die simulierten Injektorspitzen hauptsächlich Lageabweichungen in x-Richtung aufweisen. Die Schnittdarstellung weist zwei grüne Bögen auf, an denen sich besonders häufig Geometrie befindet. Entlang der z-Achse besitzt das Bauteil nahezu keine Abweichungen, so dass die jeweiligen Voxel hier blau bzw. rot eingefärbt sind. Weiterhin lässt sich erkennen, dass sich die räumlichen Schwankungen entlang der positiven y-Richtung verringern. Die Abnahme der Schwankungen entspricht dem erwarteten Verhalten, da der Injektor im oberen Bereich (hohe y-Werte) seine Lagerung besitzt.

Hier zeigt sich ein großer Vorteil des volumenbasierten Ansatzes: Bei Untersuchung des Datensatzes durch Zweipunktmessungen würde man je nach Messposition und -richtung völlig unterschiedliche Ergebnisse erzielen: Bei Messungen in z-Richtung wären nahezu keine Abweichungen des Injektors festgestellt worden, in x-Richtung sehr starke Abweichungen. Außerdem wird das Ergebnis von der Höhe der durchgeführten Messung (y-Wert) beeinflusst. Erst durch eine Vielzahl von Punktmessungen könnte man feststellen, in welche Richtungen und in welchem Ausmaß die Injektorspitze vorzugsweise schwankt.

Zur weiteren Analyse wurden durch Voxelisierung der Bauteile als Festkörper (Solids) für Injektorspitze und umliegende Bohrung Volumendatensätze erstellt. Die Auflösung und der Auswahlbereich der Voxelisierung blieben unverändert zum vorangegangenen Beispiel. Bild 67 zeigt die Volumendarstellung des Injektors (links) und der Bohrung

(Mitte). Die Bohrung weist im Vergleich zum Injektor geringere räumliche Schwankungen auf.

Für die Baugruppe ist funktional relevant, dass die Injektorspitze die umliegende Wand der Bohrung nicht berührt. Simulierte Kollisionen würden in der Realität zu einer Verbiegung der Injektorspitze an der Bohrungswand und damit zu Fehlfunktionen führen. Zur Erkennung von Kollisionen wurde anhand der beiden Solid-Volumendatensätze aus Bild 67 ein Schnittvolumen berechnet. Da sich die beiden Zylinder des untersuchten Baugruppenausschnitts in ihrer Lage gegenseitig nicht beeinflussen, konnte das in Kapitel 6.3.4 beschriebene Verfahren zur Kollisionsbestimmung von unabhängigen Baugruppen verwendet werden.

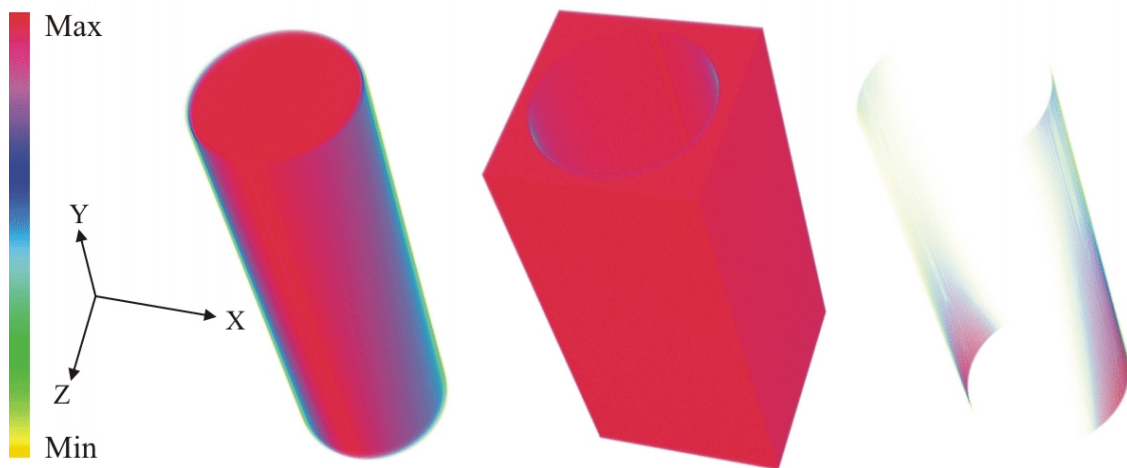


Bild 67: Kollisionsbestimmung zwischen Injektor und Bohrung für jeweils 10.000 Bauteilvarianten
 Durch die Kollisionsdarstellung (Bild 67, rechts) wird deutlich, dass entlang der gesamten Bohrungswand Bereiche mit potentiellen Überschneidungen existieren. Weiterhin lässt sich erkennen, dass Kollisionen vor allem in Richtung der x-Achse, insbesondere in positiver Richtung, auftreten. Die höchste festgestellte Kollisionswahrscheinlichkeit eines Voxels betrug 0,066% am unteren Ende der Bohrungswand. Entlang der positiven y-Achse nimmt die Kollisionswahrscheinlichkeit stark ab. Dieses Ergebnis ist konsistent mit dem in Bild 66 dokumentierten räumlichen Schwankungsverhalten der Injektorspitze.

Die Kollisionsanalyse zeigt, dass die geforderte Eigenschaft von Injektor und Bohrung nicht von allen Varianten eingehalten wird. In einigen Bereichen existieren Überschneidungen der aufsummierten Volumen. In der Realität würde dies zu einer Auflage und Verbiegung des Injektors an der Bohrungswand führen. Bei einer konventionellen Analyse durch Zweipunktmessungen wären die Kollisionen nur detektiert worden, wenn eine Messung von Injektor zur Bohrung in x-Richtung definiert worden wäre. Bei einer Abstandsmessung in z-Richtung wäre keine Kollision

feststellbar. Die geeignete Verteilung der Messpunkte gestaltet sich je nach Problemstellung kompliziert und aufwändig und bedeutet stets einen Mehraufwand für den Analyseexperten. Die Vielzahl von unabhängigen statistischen Kenngrößen der Messstellen erschwert die Interpretation der Analyseergebnisse durch den Produktentwickler.

Alternativ zu Messungen könnte eine paarweise Kollisionsberechnung der Zylinder durchgeführt werden, beispielsweise durch featurebasierte Analyse. Um die paarweise Kollisionswahrscheinlichkeit der Bauteile zu erhalten, wären einhundert Millionen Intersektionstests zwischen Zylindern nötig. Schwierig und zeitaufwändig wäre vor allem die Auswertung der Ergebnisse durch sequentielle Betrachtung aller daraus resultierenden paarweise schneidenden Zylinder. Die Volumenvisualisierung der Kollisionen fasst hingegen unter Beibehaltung der Bauteilgeometrie alle Intersektionstests in einer einheitlichen, farblich kodierten Darstellung möglicher Kollisionsbereiche zusammen. Dadurch sind die Ergebnisse der Toleranzanalyse für den Produktentwickler besser verständlich und die Lösungsfindung somit vereinfacht.

6.3.6 Volumenvisualisierung: Zusammenfassung und Ausblick

In Kapitel 6.3 wurde ein neu entwickeltes Verfahren zur statistischen Analyse von Bauteilabweichungen durch Summierte Voxelisierung und Volumenvisualisierung dargestellt. Dieses Verfahren unterstützt die Produktentwickler bei der geometrischen Interpretation der Auswirkungen von Toleranzen. Die Zusammenfassung der räumlichen Schwankungen in einem einzigen Datensatz eröffnet die Möglichkeit, große Datenmengen (z. B. die Gesamtheit von Monte-Carlo-Versuchsergebnissen) anschaulich darzustellen und zu analysieren. Anders als bei der verbreiteten statistischen Analyse durch Zweipunktmessungen können ganze Bereiche der Bauteilgeometrie untersucht werden. Mit aktueller PC-Hardware lässt sich somit das komplexe Schwankungsverhalten nichtidealer Baugruppen interaktiv betrachten. Dies verbessert das Verständnis für das Zusammenwirken der verschiedenen Toleranzzonen analysierter Baugruppen wesentlich. Die Anwendung des Verfahrens und der praktische Nutzen wurden am Beispiel abweichungsbehafteter Dreiecksnetze und Regelgeometrie (Zylindern) demonstriert. Es konnte gezeigt werden, dass das Verfahren abweichungsbedingte Probleme (z. B. Kollisionen) aufdeckt, die bei konventioneller Analyse durch Abstandsmessungen übersehen werden könnten. Die Ergebnisse der auf Abstandsmessungen basierenden Verfahren sind stark von der Definition der Messstellen abhängig. Beim entwickelten volumenbasierten Verfahren hingegen muss

der Anwender nicht a priori wissen, in welchem Bereich geometrische Probleme zu erwarten sind. Es kann interaktiv ein Volumendatensatz analysiert werden, der die gesamte Geometrie eines Auswahlbereichs repräsentiert. Neben der Feststellung der Kollisionswahrscheinlichkeit wird zusätzlich die genaue Lage und geometrische Wahrscheinlichkeitsverteilung bestimmt.

Auf diese Weise wird die Interpretation von Toleranzsimulationsergebnissen verbessert, wobei der Visualisierungsansatz sowohl für etablierte Monte-Carlo-Simulationen als auch für geometrieorientierte Simulation (siehe Kapitel 4) nutzbar ist.

Durch Fortschritte in der Computertechnik, insbesondere der Grafikhardware, können in Zukunft größere Bereiche der Bauteile mit höheren Auflösungen verarbeitet werden. Weiterhin ließen sich durch die Entwicklung von Anti-Aliasing Techniken für die Summierte Voxelisierung die in den gezeigten Volumenbildern erkennbaren Treppenstufeneffekte verringern.

7 Zusammenfassung und Ausblick

Die Vergabe von Toleranzen in der Produktentwicklung hat großen Einfluss auf zahlreiche Produkteigenschaften. Aus Sicht der Konstruktion sind möglichst enge Toleranzen wünschenswert, um eine sichere Funktionserfüllung und ggf. ein ästhetisches Aussehen zu gewährleisten. Aus Sicht der Fertigung sind enge Toleranzen jedoch nachteilig, da sie hohe Fertigungskosten aufgrund von aufwändigeren Schritten der Herstellung, Nachbearbeitung und Produktprüfung nach sich ziehen. Aufgabe der Produktentwicklung ist es daher, bei der Toleranzbestimmung einen sinnvollen Kompromiss einzugehen.

Da sich aus der Toleranzdefinition komplexe räumliche Auswirkungen ergeben, lassen sich nur einfache Baugruppen mit linearen Zusammenhängen händisch analysieren. Um nicht auf Erfahrungswissen und reale Prototypen angewiesen zu sein, werden in der Industrie momentan ausschließlich statistische Simulationsprogramme verwendet. Die Bauteilgeometrie und die Montagevorgänge werden in den verbreiteten Systemen stark vereinfacht modelliert, wodurch sich viele in der Realität auftretende Probleme nicht abbilden und aufdecken lassen.

Im Rahmen dieser Arbeit konnte gezeigt werden, wie sich die Auswirkungen toleranzbedingter Abweichungen anhand von detaillierten Geometriemodellen durchgängig simulieren und auswerten lassen. Nach der Beschreibung der von mir entwickelten Verfahren wurde die praktische Anwendbarkeit jeweils an Beispielen demonstriert.

Die Analyse einer tolerierten Baugruppe beginnt mit dem Erzeugen von virtuellen nichtidealen Einzelteilen innerhalb der Toleranzgrenzen. Je nachdem, welche Informationen über verwendete Einzelteile im Entwicklungsprozess vorliegen, können zufällige Bauteilvarianten innerhalb der Toleranzgrenzen oder Bauteilvarianten aus einer stochastischen Simulation der Fertigung als Eingangsdaten genutzt werden. Durch einen Vergleich verschiedener Möglichkeiten zur Geometriebeschreibung wurde festgestellt, dass Dreiecksnetze als universelle Form der Oberflächenmodellierung für den gewählten Simulationsansatz am besten geeignet sind.

Aufgrund der Einzelteilabweichungen wird eine Neupositionierung der einzelnen Bauteile zueinander notwendig, um die Einzelteile zu nichtidealen Baugruppen zusammenzusetzen. Dabei sind einige Plausibilitätskriterien einzuhalten: Die Bauteilvarianten dürfen sich beispielsweise nicht durchdringen und müssen nach dem

Positionierungsschema der Montage zueinander ausgerichtet sein. Diese als „Relative Positionierung“ bezeichnete Aufgabe wurde im Rahmen der Dissertation durch die Entwicklung eines Frameworks zur heuristischen Optimierung gelöst. Es wurde gezeigt, dass auf diese Weise die Positionierung flexibel an unterschiedliche Zielsetzungen angepasst werden kann. Somit sind mit diesem entwickelten Verfahren die Produktentwickler erstmals in der Lage, das reale Positionierungsschema der Bauteile einfach im Simulationsmodell abzubilden. Der Modellierungsaufwand konnte im Vergleich zu verbreiteten Toleranzanalysesystemen deutlich verringert werden. Zusätzlich sind durch die detaillierte Geometrieabbildung umfangreichere und aussagekräftigere Auswertungen möglich.

Da die Relative Positionierung ausschließlich die Endposition der Bauteile berechnet, war es darüber hinaus erforderlich, Verfahren zu entwickeln, die den Montagepfad auf Kollisionsfreiheit untersuchen. Dazu wurden Algorithmen der automatischen Pfadplanung angepasst und erweitert. Die im Rahmen dieser Arbeit entwickelten Verfahren sind nun in der Lage, eine große Anzahl von Bauteilvarianten in kurzer Zeit auf ihre Montierbarkeit hin zu untersuchen. Somit lässt sich die automatische Pfadplanung auch für statistische Analysen einsetzen. Dies war ohne die durchgeführten Modifikationen aufgrund des Rechenaufwands in der Praxis bisher nicht durchführbar.

Zur Analyse der sich ergebenden nichtidealen Baugruppen wurden mehrere Methoden konzipiert und umgesetzt. Anhand der implementierten Kopplung von Echtzeit-Physiksimulation und haptischem Interaktionsgerät kann der Nutzer die virtuelle Geometrie mit hoher Auflösung abtasten. Geometrieabweichungen, die visuell nicht erkennbar sind, können mit dem entwickelten System wahrgenommen werden. Zur graphischen Datenanalyse wurden zwei Verfahren vorgestellt: Automatische Abstandsmessungen visualisieren Abstände zwischen nichtidealen Bauteilen. Die Abstandsmessungen können zur Analyse einzelner simulierter Baugruppen verwendet werden. Die neu entwickelte Summierte Voxelisierung ermöglicht somit, die statistischen Ergebnisse von Toleranzuntersuchungen unter Beibehaltung der Geometrie in nur einem einzigen Datensatz zusammenzufassen. Durch Volumenvisualisierung ist es im Anschluss möglich, die fertigungsbedingten räumlichen Schwankungen für die Produktentwickler verständlich darzustellen. Die über Summierte Voxelisierung ermittelbaren Ergebnisse wie Kollisions- und Aufenthaltswahrscheinlichkeiten im Raum ließen sich bisher mit keinem am Markt verfügbaren Toleranzanalysesystem bestimmen und stellen somit einen deutlichen Fortschritt in der Toleranzanalyse dar.

Die entwickelten Verfahren zeigen ihren Nutzen darin, frühzeitig die geometrischen Auswirkungen von Bauteilabweichungen vorherzusagen. So lassen sich mögliche Probleme (z. B. Kollisionen bei der Montage oder ästhetisch bzw. funktional nicht zulässige Spaltmaße) erkennen, bevor der erste physikalische Prototyp gefertigt wird. Durch nichtideale virtuelle Prototypen können somit Toleranzänderungen und konstruktive Anpassungen vorgezogen werden, d.h. die Gesamtkosten werden reduziert und die Qualität des verkauften Produkts wird gesteigert.

Die vorgeschlagene Vorgehensweise zur Vorhersage von Toleranzeffekten unterscheidet sich deutlich von den etablierten, stark vereinfachenden Simulationsverfahren. Da es möglich ist auf detaillierter Geometrie rechenintensive Operationen (z. B. Kollisionserkennung) durchzuführen, ist der Rechenaufwand ungleich höher. Gerechtfertigt ist dieser Aufwand dadurch, dass damit realistischere nichtideale Baugruppen erzeugt, analysiert und visualisiert werden. Diese besitzen einen höheren Informationsgehalt als die bisherigen statistischen Simulationsergebnisse. Geometrische Probleme werden aufgedeckt, die ansonsten ggf. nicht auffallen würden. Weiterhin vereinfacht sich für den Produktentwickler der Modellierungsaufwand der Toleranzanalyse, da die Bauteile ohne Abstraktion miteinander kombiniert werden und vorab keine Definition von Messstellen notwendig ist.

Um die geometrieorientierte Toleranzanalyse als praktikables Werkzeug der Produktentwicklung einzusetzen, müssen nichtideale Bauteilvarianten realistisch simuliert werden, wie beispielsweise in [115] für das Fertigungsverfahren „Tiefziehen“ gezeigt wurde. Um weitere Fertigungsverfahren zu integrieren, ist zusätzlicher Forschungsaufwand nötig. Hilfreich wäre es außerdem, basierend auf der Toleranzdefinition realitätsnahe nichtideale Einzelteile zu generieren, selbst wenn das Fertigungsverfahren noch nicht festgelegt wurde. Es existieren zwar erste Ansätze auf diesem Gebiet, jedoch ist es noch nicht möglich, für die Kombination beliebiger Toleranzzonen Einzelteilvarianten zu erstellen, die den Toleranzbereich vollständig ausnutzen bzw. ein realistisches Streuungsverhalten aufweisen.

Bei der Montage von flächigen Bauteilen oder Kunststoffteilen treten oft Verformungen auf oder sind zum Teil für die Montage sogar notwendig. Daher sollte künftig die Modellierung und Simulation von nachgiebigen Bauteilen in der Toleranzanalyse mit detaillierter Bauteilgeometrie erforscht werden. Um statistische Aussagen treffen zu können, müssen die zu entwickelnden Verfahren sowohl entstehende Kräfte als auch

das Materialverhalten der Bauteile korrekt abbilden und dennoch nach angemessener Rechenzeit verlässliche Ergebnisse liefern.

8 Literaturverzeichnis

- [1] Akenine-Möller, T.: Fast 3D triangle-box overlap testing. In: SIGGRAPH 2005 ACM SIGGRAPH Courses, New York: ACM 2005.
- [2] Akenine-Möller, T.; Haines, E.: Real-time rendering. 2nd ed. Natick Mass.: AK Peters 2002.
- [3] Albert, S.: Erweiterung der Interaktionsmöglichkeiten einer Haptikanwendung., Studienarbeit, Friedrich-Alexander-Universität Erlangen-Nürnberg 2009.
- [4] Amato, N.M.; Wu, Y.: A Randomized Roadmap Method for Path and Manipulation Planning. In: Proceedings of the 1996 IEEE International Conference on Robotics and Automation 1996, S. 113–120.
- [5] ASME Standard Y14.5.1M: Dimensioning and Tolerancing 1994. American Society of Mechanical Engineers. New York 1994.
- [6] Ameta, G.; Davidson, J.K.; Shah, J.J.: Using Tolerance-Maps to generate Frequency Distributions of Clearance for Pin-Hole Assemblies. In: ASME Design Automation Conference, Philadelphia 2006.
- [7] Bach, J.; Guibas, L.J.; Hsu, D.; Nguyen, A.T.: Disconnection Proofs for Path Planning. In: International Conference on Robotics and Automation 2001, S. 1765–1772.
- [8] Ballot, E.; Bourdet, P.: A computation method for the consequences of geometric errors in mechanisms. In: Geometric Design Tolerancing: Theories, Standards and Applications 1998, S. 197–207.
- [9] Ballot, E.; Bourdet, P.: A mathematical model of geometric errors in the case of specification and 3D control of mechanical parts. In: Advanced mathematical and computational tools in Metrology IV: World Scientific Publishing Company 2000, S. 11–20.
- [10] Baumgärtner, T.: Test und Weiterentwicklung einer Haptik-Anwendung., Bachelorarbeit, Friedrich-Alexander-Universität Erlangen-Nürnberg 2007.
- [11] Bayazit, O.B.; Song, G.; Amato, N.M.: Enhancing Randomized Motion Planners: Exploring with Haptic Hints. In: Autonomous Robots: Kluwer Academic Publishers 2001.
- [12] Besl, P.J.; McKay, N.D.: A Method for Registration of 3D Shapes. In: IEEE Transactions on pattern analysis and machine intelligence 1992, S. 239–256.
- [13] Blaurock, J.E.: Haptische Wahrnehmung rechnerinterner Freiformflächen., Dissertation, Universität Kaiserslautern 2004.

-
- [14] Bourdet, P.; Ballot, E.: Geometric behavior laws for computer aided tolerancing. In: Proceedings of the 4th CIRP Design Seminar on Computer Aided Tolerancing, S. 119–131.
- [15] Chase, K.W.: Tolerance Analysis of 2-D and 3-D Assemblies, Brigham Young University 1999.
- [16] Chase, K.W.; Gao, J.; Magleby, S.O.; Sorensen, C.D.: Including geometric feature variations in tolerance analysis of mechanical assemblies. IIE Transactions, Band 28 (1996.) 10, S. 795–807.
- [17] Clement, A.; Desrochers, A.; Riviere, A.: Theory and practice of 3D tolerancing for assembly. In: CIRP Computer Aided Tolerancing 1991.
- [18] Clement, A.; Riviere, A.; Serre, P.; Valade, C.: The TTRSs: 13 constraints for dimensioning and tolerancing. In: Geometric Design Tolerancing: Theories, Standards and Applications 1998, S. 122–131.
- [19] Davidson, J.K.; Mujezinovic, A.; Shah, J.J.: A New Mathematical Model for Geometric Tolerances as Applied to Round Faces. Journal of mechanical design, Band 124 (2002.) 4, S. 609–622.
- [20] Davidson, J.K.; Shah, J.J.: The method of isosceles triangulation validated for constructing T-Maps for size of parts having round and rectangular cross-sections. In: CIRP Computer Aided Tolerancing Conference 2009.
- [21] D'Errico, J.R.; Zaino, N.A.: Statistical Tolerancing Using a Modification of Taguchi's Method. In: Technometrics: American Statistical Association and American Society for Quality 1988, S. 397–405.
- [22] Desrochers, A.; Riviere, A.: A matrix approach to the representation of tolerance zones and clearances. In: The International Journal of Advanced Manufacturing Technology, London: Springer 1997, S. 630–636.
- [23] DIN 4760: Gestaltabweichungen; Begriffe, Ordnungssystem. 1982. DIN Deutsches Institut für Normung. Berlin: Beuth-Verlag 1982.
- [24] DIN ISO 8015: Technische Zeichnungen; Tolerierungsgrundsatz. 1986. DIN Deutsches Institut für Normung. Berlin: Beuth-Verlag 1986.
- [25] DIN 7167: Zusammenhang zwischen Maß-, Form- und Parallelitätstoleranzen; Hüllbedingung ohne Zeichnungseintragung. 1987. DIN Deutsches Institut für Normung. Berlin: Beuth-Verlag 1987.
- [26] DIN ISO 2768: Allgemeine Toleranzen. Teil 1: Toleranzen für Längen- und Winkelmaße ohne einzelne Toleranzeintragung, Teil 2: geometrische Toleranzen für Konstruktionsmerkmale ohne einzelne Toleranzeintragung. 1989. DIN Deutsches Institut für Normung. Berlin: Beuth-Verlag 1989.
- [27] DIN ISO 286: ISO-System für Grenzmaße und Passungen. 1990. DIN Deutsches Institut für Normung. Berlin: Beuth-Verlag 1990.

-
- [28] DIN ISO 1101: Technische Zeichnungen; Form- und Lagetolerierung; Form-, Richtungs-, Orts- und Lauftoleranzen; Allgemeines, Definitionen, Symbole, Zeichnungseintragungen. 2004. DIN Deutsches Institut für Normung. Berlin: Beuth-Verlag 2004.
- [29] Drieux, G.; Guillaume, F.; Leon, J.C.; Chevassus, N.: SAMIRA: A Platform for Virtual Maintenance Simulation with Haptic Feedback incorporating a model preparation process. In: Proceedings of Virtual Concept, Biarritz 2005, S. 150–158.
- [30] Eisemann, E.; Decoret, X.: Single-pass GPU Solid Voxelization and Applications. In: GI '08: Proceedings of Graphics Interface 2008: Canadian Information Processing Society 2008, S. 73–80.
- [31] Ericson, C.: Real-time collision detection. Amsterdam, Boston: Elsevier 2005.
- [32] Ferré, E.; Laumond, J.P.: An Iterative Diffusion Algorithm for Part Disassembly. In: Proceedings of the 2004 IEEE International Conference on Robotics and Automation, New Orleans 2004, S. 3149–3154.
- [33] Fogel, L.J.; Owens, A.J.; Walsh, M.J.: Artificial intelligence through simulated evolution. New York: Wiley and Sons 1966.
- [34] Foley, J.D.; VanDam, A.; Feiner, S.K.: Computer graphics: *Principles and practice*. 2nd ed. Reading Mass.: Addison-Wesley 1995.
- [35] Gao, J.; Chase, K.W.; Magleby, S.P.: Generalized 3-D tolerance analysis of mechanical assemblies with small kinematic adjustments. IIE Transactions, Band 30 (1998.) 4, S. 367–377.
- [36] Gao, J.; Chase, K.W.; Magleby, S.P.: Global coordinate method for determining sensitivity in assembly tolerance analysis, Brigham Young University 1998.
- [37] Gliniorz, S.: Anwendertreffen Toleranzanalyse der Bosch GmbH 2008.
- [38] Gnezdilov, A.: Geschwindigkeitssteigerung eines Optimierungsframeworks zur relativen Positionierung, Diplomarbeit. Erlangen 2007.
- [39] Gnezdilov, A.; Wittmann, S.; Helwig, S.; Kókai, G.: Acceleration of a Relative Positioning Framework. In: International Journal of Computational Intelligence Research, MIR Labs(Hrsg.) 2009, S. 130–140.
- [40] Goldberg, D.E.: Genetic algorithms in search, optimization, and machine learning. Addison-Wesley 1989.
- [41] Gossard, D.C.; Zuffante, R.P.; Sakurai, H.: Representing Dimensions, Tolerances and Features in MCAE Systems. IEEE Computer Graphics and Applications, Band 8 (1988.) 2, S. 51–59.
- [42] Gottschalk, S.; Lin, M.C.; Manocha, D.: OBBTree: a hierarchical structure for rapid interference detection. In: SIGGRAPH '96: Proc. of the 23rd annual

- conference on Computer graphics and interactive techniques: ACM 1996, S. 171–180.
- [43] Grätz, J.-F.: Handbuch der 3D-CAD-Technik: *Modellierung mit 3D-Volumensystemen*. Berlin: Siemens AG Abt. Verl. 1989.
- [44] Gropp, W.; Lusk, E.: User's Guide for MPI 1996
- [45] Gupta, S.; Turner, J.U.: Variational Solid Modeling for Tolerance Analysis. IEEE Computer Graphics and Applications, Band 13 (1993.) 3, S. 64–74.
- [46] Hart P.E., N.N.J.R.B.: A Formal Basis for the Heuristic Determination of Minimum Cost Paths. In: IEEE Transactions on Systems Science and Cybernetics 1968, S. 100–107.
- [47] Hong, Y.S.; Chang, T.C.: A comprehensive review of tolerancing research. In: International Journal of Production Research: Taylor&Francis 2002, S. 2425–2459.
- [48] Hoschek, J.; Lasser, D.: Grundlagen der geometrischen Datenverarbeitung. Stuttgart: Teubner 1989.
- [49] Hsu, D.; Kavradi, L.E.; Latombe, J.-C.; Motwani, R.; Sorkin, S.: On Finding Narrow Passages with Probabilistic Roadmap Planners. In: Robotics: The Algorithmic: A. K. Peters, S. 141–154.
- [50] Humienny, Z.; Weckenmann, A.; Bialas, S.; Osanna, P.; Tampre, M.; Blunt, W.; Jakubiec, W.: Geometrische Produktspezifikation (GPS): *Kurs für Technische Universitäten*. Lehrstuhl für Qualitätsmanagement und Fertigungsmesstechnik 2001.
- [51] IEEE 754: IEEE Standard for Floating-Point Arithmetic 2008. IEEE 2008.
- [52] Iglberger, K.: Software Design of a Massively Parallel Rigid Body Framework., Dissertation, Friedrich-Alexander-Universität Erlangen-Nürnberg 2010.
- [53] Iglberger, K.; Rüde, U.: Massively Parallel Rigid Body Dynamics Simulations. In: Computer Science - Research and Development 2009.
- [54] Iglberger, K.; Rüde, U.: The PE Rigid Multi-Body Physics Engine, Friedrich-Alexander-Universität Erlangen-Nürnberg 2009.
- [55] Inui, M.; Miura, M.; Kimura, F.: Relative positioning of assembled parts with small geometric deviations by using hierarchically approximated configuration space. Proc. of the 1995 IEEE Int. Conf. on Robotics and Automation, Band 2 (1995.), S. 1605-1612.
- [56] Jorden, W.: Form- und Lagetoleranzen: *Handbuch für Studium und Praxis*. 3rd ed. München: Hanser 2005.

-
- [57] Juster, N.; Fitchie, M.; Taylor, S.; Dew, P.M.; Maxfield, J.; Zhao, J.: Visualizing the Impact of Tolerances On Cosmetic Product Quality. In: Proceedings of the 13th International Conference on Engineering Design (ICED 2001), Glasgow 2001.
- [58] Juster, N.; Maxfield, J.; Dew, P.M.; Taylor, S.; Fitchie, M.; Ion, W.J.; Zhao, J.; Thompson, M.: Predicting Product Aesthetic Quality Using Virtual Environments. In: Journal of Computing and Information Science in Engineering 2001.
- [59] Kavraki, L.E.; Svestka, P.; Latombe, J.-C.; Overmars, M.H.: Probabilistic Roadmaps for Path Planning in High-Dimensional Configuration Spaces. In: IEEE Transactions on Robotics and Automation 1996, S. 566–580.
- [60] Kennedy, J.; Eberhart, R.C.: Particle Swarm Optimization. In: Proc. of IEEE Int. Conf. on Neural Networks: Piscataway, NJ 1995, S. 1942–1948.
- [61] Kennedy, J.; Eberhart, R.C.: Swarm Intelligence. Morgan Kaufmann Academic Press 2001.
- [62] Kevin Kreeger, V.I.: CT scan of a backpack., verfügbar unter <http://www.volvis.org/>, Stand: 06/2010.
- [63] Kirkpatrick, S.; Gelatt, C.D.; Vecchi M.P.: Optimization by Simulated Annealing. In: Science 1983, S. 671–680.
- [64] Koch, A.: Entwicklung einer Methode zur Visualisierung der Auswirkungen von Form- und Lagetoleranzen auf die Bauteilgestalt. Düsseldorf: VDI-Verl. 2006.
- [65] Koch, A.; Meerkamm, H.: TolVis – Visualisierung von möglichen Gestaltabweichungen resultierend aus Form- und Lageabweichungen. In: Design for X - Beiträge zum 15. Symposium, Neukirchen, Germany 2004.
- [66] Koch, A.; Meerkamm, H.: TolVis - Visualization of possible Shape Variations resulting from Shape and Position Tolerances. In: Proceedings of the 15th International Conference on Engineering Design (ICED 2005), Melbourne 2005.
- [67] Koza, R.J.: Genetic programming: on the programming of computers by means of natural selection. MIT Press 1992.
- [68] Kuffner, J.J.; LaValle, S.: RRT-Connect: An efficient approach to single-query path planning. In: IEEE International Conference on Robotics and Automation 2000.
- [69] Latombe, J.-C.: Motion Planning: A Journey of Robots, Meolecules, Digital Actors, and Other Artifacts. In: The International Journal of Robotics Research, Sage Publications(Hrsg.) 1999, S. 1119–1128.
- [70] LaValle, S.: Rapidly Exploring Random Trees: A New Tool for Path Planning: *Technical Report 98-11* 1998.

- [71] LaValle, S.; Kuffner, J.J.: Rapidly-exploring random trees: Progress and prospects. In: Workshop on the Algorithmic Foundations of Robotics 2000.
- [72] LaValle, S.; Kuffner, J.J.: Randomized Kinodynamic Planning. In: The International Journal of Robotics Research 2001, S. 378–400.
- [73] LaValle, S.M.: Planning algorithms. Cambridge , New York: Cambridge University Press 2006.
- [74] Lecuyeer, A.; Kheddar, A.; Coquillart, S.; Graux, L.; Coiffet, P.: A Haptic Prototype for the Simulation of Aeronautic Mounting/Unmounting Operations. In: IEEE International Workshop on Robot-Human Interactive Communication, Paris 2001.
- [75] Lee, W.; Requicha A. A. G.: Algorithms for computing the volume and other integral properties of solids. In: Communications of the ACM 1982, S. 635–650.
- [76] Li, B.; Roy, U.: Relative positioning of toleranced polyhedral parts in an assembly. IIE Transactions, Band 33 (2001.), S. 323-336.
- [77] Liu, J.; Wilhelm, R.G.: Genetic Algorithms for TTRS tolerance analysis. In: CIRP International Seminar on Computer Aided Tolerancing 2001, S. 55–64.
- [78] Lööf, J.; Söderberg, R.; Lindkvist, L.: Visualization of motion envelope of parts and assemblies based on simulation or measurement data. In: ASME International Mechanical Engineering Congress and Exposition, Chicago 2006.
- [79] Lööf, J.; Söderberg, R.; Lindkvist, L.: Visualization of variation in early design phases: A convex hull approach. In: Proc. of the 9th International Design Conference, Dubrovnik 2006, S. 905–912.
- [80] Lorensen, W.E.; Cline, H.E.: Marching Cubes: A high resolution 3D surface construction algorithm. In: Computer Graphics Vol. 21 1987, S. 164–169.
- [81] Mannewitz, F.: Komplexe Toleranzanalysen einfach durchführen. Konstruktion. (2004.) Juli/August, S. 69 ff.
- [82] Mannewitz, F.: Baugruppenfunktions- und prozessorientierte Toleranzaufweitung. Konstruktion. (2005.) 10, S. 87–93.
- [83] Maxfield, J.; Dew, P.M.; Zhao, J.; Juster, N.; Fitchie, M.: A Virtual Environment for Aesthetic Quality Assessment of Flexible Assemblies in the Automotive Design Process. In: SAE 2002 World Congress, Detroit, Michigan 2002.
- [84] McKay, M.D.; Beckman, R.j.; Conover W.J.: A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code. In: Technometrics 1979, S. 239–245.
- [85] Meerkamm, H.; Stuppy, J.; Koch, M.: Fertigungsgerechtes Konstruieren.: *Skriptum zur Vorlesung*, Friedrich-Alexander-Universität Erlangen-Nürnberg 2009.

-
- [86] Morière, S.; Mailhe, J.; Linares, J.; Spauel, J.: Assembly method comparison including form defect. In: CIRP Computer Aided Tolerancing Conference 2009.
- [87] Mujezinovic, A.; Davidson, J.K.; Shah, J.J.: A New Mathematical Model for Geometric Tolerances as Applied to Polygonal Faces. *Journal of mechanical design*, Band 126 (2004.) 3, S. 504–518.
- [88] Nigam, S.D.; Turner, J.U.: Review of statistical approaches to tolerance analysis. In: *Computer Aided Design*: Elsevier 1995, S. 6–15.
- [89] Nissoux, C.; Simeon, T.; Laumond, J.P.: Visibility Based Probabilistic Roadmaps. In: *Proceedings of the IEEE International Conference on Intelligent Robots and Systems 1999*, S. 1316–1321.
- [90] Ostrovsky-Berman, Y.; Joskowicz, L.: Tolerance envelopes of planar mechanical parts with parametric tolerances. *Computer-aided design*, Band 37 (2005.) 5, S. 531–544.
- [91] Otter, L.: Erweiterung und Integration von dynamischen Elementen in ein Haptik-System., Projektarbeit, Friedrich-Alexander-Universität Erlangen-Nürnberg 2008.
- [92] Penzkofer, F.: Entwicklung eines Softwarewerkzeugs zur Analyse abweichungsbehafteter Geometrie., Diplomarbeit, Friedrich-Alexander-Universität Erlangen-Nürnberg 2007.
- [93] Penzkofer, F.; Wittmann, S.; Winter, M.: Visual analysis methods for non-ideal assemblies. In: *Proc. of the 10th International Design Conference (DESIGN 2008)*, Dubrovnik 2008, S. 657–664.
- [94] Perret, J.: Physiksimulation und 6D Haptik. In: *3D in Motion with Haptic Experience*, 3Dims(Hrsg.), Frankfurt 2010.
- [95] Pierce, R.S.; Rosen, D.: NURBS-Based Variational Modeling as a Tool for the Analysis of Geometric Tolerances. In: *The 1997 ASME Design Engineering Technical Conference*, Sacramento 1997.
- [96] Pierce, R.S.; Rosen, D.: Simulation of Mating Between Nonanalytic Surfaces Using a Mathematical Formulation. *Journal of Computing and Information Science in Engineering*, Band 7 (2007.), S. 314–321.
- [97] Pierce, R.S.; Rosen, D.: A method for integrating form errors into geometric tolerance analysis. In: *Journal of mechanical design*, January 2008: ASME 2008, S. 11002–11012.
- [98] Prisco, U.; Giorleo, G.: Overview of current CAT systems. In: *Integrated Computer-Aided Engineering 9*, IOS Press(Hrsg.) 2002, S. 373–387.
- [99] Rechenberg, I.: Evolutionsstrategie □ Optimierung technischer Systeme nach Prinzipien der biologischen Evolution. Frommann-Holzboog 1973.

-
- [100] Requicha, A.A.G.: Toward a Theory of Geometric Tolerancing. *International Journal of Robotics Research*, Band 2 (1983.) 4, S. 45–60.
- [101] Rezk-Salama, C.: Volume rendering techniques for general purpose graphics hardware., Dissertation, Friedrich-Alexander-Universität Erlangen-Nürnberg 2002.
- [102] Roy, U.; Li, B.: Representation and interpretation of geometric tolerances for polyhedral objects. II. Size, orientation and position tolerances. *Computer-aided design*, Band 31 (1999.) 4, S. 273–285.
- [103] Salomons, O.W.; Haalboom, F.J.; Jonge Poerink H.J.; Slooten, F.; Houten, F.J.A.M.; Kals, H.J.J.: A computer aided tolerancing tool II: Tolerance analysis. *Computers in Industry*, Band 31 (1996.) 2, S. 175–186.
- [104] Salomons, O.W.; Jonge Poerink H.J.; Haalboom, F.J.; Slooten, F.; Houten, F.J.A.M.; Kals, H.J.J.: A computer aided tolerancing tool I: Tolerance specification. *Computers in Industry*, Band 31 (1996.) 2, S. 161–174.
- [105] Shah, J.J.; Ameta, G.; Shen, Z.; Davidson, J.K.: Navigating the Tolerance Analysis Maze. In: *Computer Aided Design & Applications 2007*, S. 705–718.
- [106] Shoemake, K.: Animating rotation with quaternion curves. In: *International Conference on Computer Graphics and Interactive Techniques: Proceedings of the 12th annual conference on Computer graphics and interactive techniques*, ACM(Hrsg.), New York 1985, S. 245–254.
- [107] Singh, G.; Ameta, G.; Davidson, J.K.; Shah, J.J.: Worst-Case tolerance analysis of self-aligning coupling assembly using tolerance maps. In: *CIRP Computer Aided Tolerancing Conference 2009*.
- [108] Smith, R.: The Open Dynamics Engine User Guide. <http://www.ode.org/ode-docs.html> 2009.
- [109] Söderberg, R.; Lindkvist, L.: Automated Seam Variation and Stability Analysis for Automotive Body Design. In: *7th CIRP International Seminar on Computer Aided Tolerancing*, Cachan 2001, S. 355–364.
- [110] Söderberg, R.; Lindkvist, L.; Carlson, J.: Virtual geometry assurance for effective product realization. In: *1st Nordic Conference on Product Lifecycle Management*, Göteborg 2006, S. 75–86.
- [111] Sodhi, R.; Turner, J.U.: Relative Positioning of variational part models for design analysis. *Computer-aided design*, Band 26 (1994.) 5, S. 366–378.
- [112] Stefano, P.D.; Angelo, L.D.: Geometric Modeling of Mechanical Parts affected by Shape Error. In: *Proc. of the 8th International Design Conference (DESIGN 2004)*, Dubrovnik 2004, S. 157–162.
- [113] Stockinger, A.: Computer Aided Robust Design – Verknüpfung rechnerunterstützter Entwicklung und virtueller Fertigung als Baustein des

- Toleranzmanagements (als Dissertation angenommen)., Dissertation, Friedrich-Alexander-Universität Erlangen-Nürnberg 2010.
- [114] Stockinger, A.; Meerkamm, H.: Concept for the integration of manufacturing simulations into tolerance analysis. In: Proceedings of the 17th International Conference on Engineering Design (ICED 2009), Stanford, CA 2009, S. 181–192.
- [115] Stockinger, A.; Wittmann, S.; Martinek, M.; Meerkamm, H.; Wartzack, S.: Virtual Assembly Analysis: Standard Tolerance Analysis compared to Manufacturing Simulation and Relative Positioning. In: Proc. of the 11th International Design Conference (DESIGN 2010), Dubrovnik 2010.
- [116] Stoll, T.: Generieren von nichtidealer Geometrie. In: Symposium Design for X, Meerkamm, H.(Hrsg.), Neukirchen, Germany 2006, S. 314–321.
- [117] Stoll, T.; Paetzold, K.: Gap and flush visualization of virtual nonideal prototypes. In: Proc. of the 10th International Design Conference (DESIGN 2008), Dubrovnik 2008.
- [118] Stoll, T.; Wittmann, S.; Helwig, S.; Paetzold, K.: Registration of measured and simulated non-ideal geometry using optimization methods. In: Proc. of the 10th CIRP International Conference on Computer Aided Tolerancing 2007, Erlangen 2007.
- [119] Stoll, T.; Wittmann, S.; Meerkamm, H.: Tolerance Analysis with detailed Part Modeling including Shape Deviations. In: Proceedings of the 11th CIRP International Conference on Computer Aided Tolerancing, Annecy 2009.
- [120] Stoll, T.; Wittmann, S.; Meerkamm, H.: Tolerance Analysis with detailed Part Modeling. In: Product Life-Cycle Management: Geometric Variations: ISTE Wiley 2010.
- [121] Stone, R.J.: Haptic Feedback: A Potted History, From Telepresence to Virtual Reality. In: Proceedings of the First International Workshop on Haptic Human-Computer Interaction 2000, S. 1–16.
- [122] Stuppy, J.; Meerkamm, H.; Wartzack, S.: Statistical tolerance analysis and result visualisation for systems in motion. In: Proceedings of the 11th International Design Conference DESIGN 2010, Dubrovnik 2010, S. 1431–1440.
- [123] Sußner, G.: Methoden zur Erzeugung und Darstellung von tessellierten Daten im Kontext der interaktiven virtuellen Qualitätskontrolle in der Fahrzeugentwicklung., Dissertation, Friedrich-Alexander-Universität Erlangen-Nürnberg 2008.
- [124] Sutherland, I.E.: Sketchpad: *A man-machine graphical communication system*. New York: Garland Pub. 1963.
- [125] Taguchi, G.: Performance Analysis Design. In: International Journal of Production Research 1978, S. 512–530.

-
- [126] The Virtual Reality Modeling Language:ISO/IEC 14772., verfügbar unter <http://www.web3d.org/x3d/specifications/vrml/ISO-IEC-14772-VRML97/>, Stand: 06/2009.
- [127] UNC Research Group on Modeling Physically-Based Simulation and Applications: PQP - A Proximity Query Package., verfügbar unter <http://www.cs.unc.edu/~geom/SSV/>, Stand: 05/2009.
- [128] Watt, A.; Watt, M.: Advanced animation and rendering techniques: *Theory and Practice*. New York: ACM 1992.
- [129] Westermann, R.; Ertl, T.: Efficiently Using Graphics Hardware in Volume Rendering Applications. In: International Conference on Computer Graphics and Interactive Techniques 1998, S. 169–177.
- [130] Whitney, D.E.; Gilbert, O.L.; Jastrzebski, M.: Representation of geometric variations using matrix transforms for statistical tolerance analysis in assemblies. Research in Engineering Design, Band 6 (1994.) 4, S. 191–210.
- [131] Wickman, C.; Söderberg, R.: Increased Concurrency Between Industrial and Engineering Design using CAT Technology Combined with Virtual Reality. Journal of Concurrent Engineering Research and Applications. (2007.) 11, S. 7–15.
- [132] Wickman, C.; Söderberg, R.: Perception of gap and flush in virtual environments. Journal of Engineering Design, Band 18 (2007.) 2, S. 175–193.
- [133] Wickman, C.; Söderberg, R.; Lindkvist, L.: Toward non-nominal virtual geometric verification by combining VR and CAT technologies. In: CIRP International Seminar on Computer Aided Tolerancing 2001, S. 233–242.
- [134] Wittmann, S.: Visualisierung von abweichungsbehafteten Flächen in Virtual Reality., Studienarbeit, Friedrich-Alexander-Universität Erlangen-Nürnberg 2004.
- [135] Wittmann, S.; Stockinger, A.: Volumenvisualisierung von abweichungsbehafteter Geometrie. In: Symposium Design for X, Meerkamm, H.(Hrsg.), Neukirchen, Germany 2006.
- [136] Wittmann, S.; Stockinger, A.; Stoll, T.; Paetzold, K.: A system for haptic tolerance analysis regarding non-ideal geometry. In: Proc. of the 10th International Design Conference (DESIGN 2008), Dubrovnik 2008, S. 729–736.
- [137] Wittmann, S.; Stoll, T.; Paetzold, K.: Volume Visualization of Geometric Deviations. In: Proceedings of the 16th International Conference on Engineering Design (ICED 2007), Paris 2007.
- [138] Wittmann, S.; Winter, M.; Paetzold, K.: Assembly Simulation for Toleranced Parts: An Adapted Path Planning Approach. In: International Conference on Engineering Design (ICED'09), Stanford, CA 2009.

-
- [139] Yamrom, B.: The Alpha Puzzle., verfügbar unter <http://parasol.tamu.edu/dsmft/benchmarks/mp/alpha/>, Stand: 05/2010.
- [140] Yershova, A.; LaValle, S.: Improving Motion Planning Algorithms by Efficient Nearest Neighbor Searching. In: IEEE Transactions on Robotics 2007, S. 151–157.
- [141] Zhang, L.; Kim, Y.J.; Manocha, D.: Efficient Cell Labelling and Path Non-existence Computation using C-obstacle Query. In: The International Journal of Robotics Research 2008, S. 1246–1257.

9 Anhang

9.1 Prozessfähigkeitsindex C_p

Der Prozessfähigkeitsindex C_p dient als Bewertungsmaß für Fertigungsprozesse und ist definiert als

$$C_p = \frac{OSG - USG}{6\sigma}.$$

σ = Standardabweichung

OSG= Obere Spezifikationsgrenze

USG= Untere Spezifikationsgrenze

Hinreichend sichere Fertigungsprozesse besitzen einen C_p -Wert größer 1, d.h. mehr als 99,7% der Ergebnisse liegen innerhalb des Bereichs $\mu \pm 3\sigma$ (bei einer Normalverteilung des Messwertes). C_p -Werte von 1,3, 1,6 und höher sind in der industriellen Fertigung üblich (siehe [56]).

9.2 Pfadvereinfachung

Die eingesetzten randomisierten Pfadplanungsalgorithmen aus Kapitel 5.1.2 liefern bei erfolgreicher Suche einen zusammenhängenden kollisionsfreien Pfad, der meist zahlreiche unnötige Zwischenpositionen enthält (siehe Bild 68, links). Um das gelieferte Ergebnis (im folgenden Text als Pfad A bezeichnet) zu einem sinnvollen Montageablauf umzuformen (Bild 68, rechts), ist eine Bereinigung erforderlich.

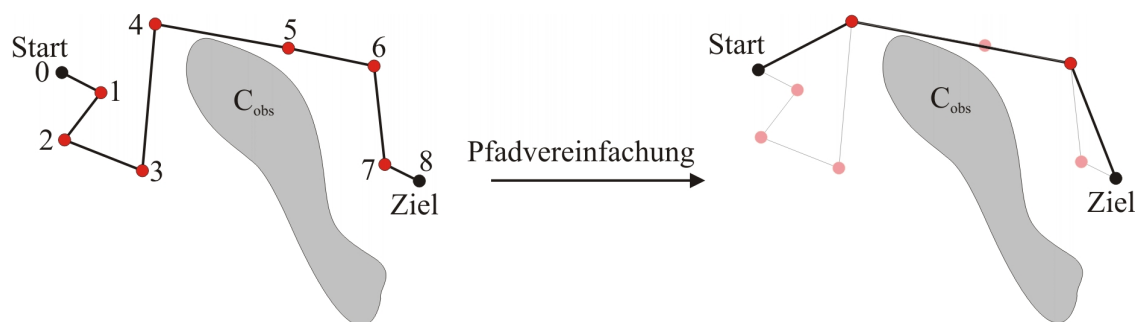


Bild 68: Reduktion eines kollisionsfreien Verbindungspfads von 9 auf 4 Zwischenpositionen

Vermieden werden sollte dabei eine erneute zeitaufwändige Pfadplanung, die zusätzliche Zwischenpositionen generiert. Der implementierte Algorithmus verwendet ausschließlich die bekannten kollisionsfreien Zwischenpositionen von Pfad A.

Realisiert wurde die Pfadvereinfachung dadurch, dass der implementierte Algorithmus von der vorgegebenen Startposition aus versucht, durch direkte Verbindung mit einer bereits bestehenden Pfadposition möglichst weit zur Zielposition zu gelangen. Im

Beispiel aus Bild 68 wird die direkte Verbindungsstrecke der Positionen 0 und 8 auf Kollisionen geprüft, dann die Verbindung 0-7, 0-6 ... , bis schließlich die kollisionsfreie Verbindung 0-4 gefunden wird. Es folgen die Verbindungsversuche 4-8, 4-7, 4-6 und 6-8. Als Ergebnis wird Pfad B mit den Knoten 0, 4, 6 und 8 zurückgeliefert. Neben der linearen Suche nach einem Sprungziel hat sich die binäre Suche als praktikabel erwiesen, insbesondere bei der Vereinfachung von sehr langen Pfaden.

9.3 MPNN, Abstandsmetriken

Die lineare Nachbarschaftssuche verwendet zur Abstandsberechnung zwischen zwei Positionen p_1 und p_2 die Metrik M_1 :

$$M_1(p_1, p_2) = \sqrt{(1-r) \cdot (\Delta\theta)^2 + r \cdot \frac{\Delta x^2 + \Delta y^2 + \Delta z^2}{d_{\max}^2}} \quad \text{Formel 10}$$

wobei

d_{\max} = Länge der Suchraumdiagonale

r = Gewichtungsfaktor der Translation

$\Delta\theta = \cos^{-1}|q_1 \bullet q_2|$ = Abstand der Quaternionen

Bei der Verwendung der Bibliothek MPNN ist die eingesetzte Metrik M_2 wie folgt definiert:

$$M_2(p_1, p_2) = \sqrt{((1-r) \cdot \Delta\theta)^2 + r^2 \cdot (\Delta x^2 + \Delta y^2 + \Delta z^2)} \quad \text{Formel 11}$$

Durch die schnelle Nachbarschaftssuche war MPNN in der Lage, Abfragen schneller zu beantworten als die lineare Suche. In manchen Experimenten generierte MPNN jedoch einen größeren Suchbaum und rief die Kollisionserkennung öfter auf. Die Erfolgswahrscheinlichkeit war mit Metrik M_1 in den Vergleichsexperimenten zur Pfadplanung meist höher als bei der Verwendung von Metrik M_2 .